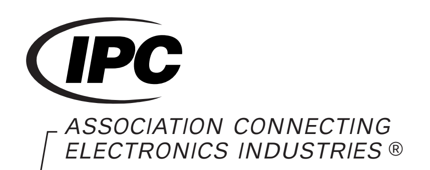


IPC-2531

SMEMA Standard Recipe File Format Specification

IPC-2531

March 1999

A standard developed by IPC

2215 Sanders Road, Northbrook, IL 60062-6135
Tel. 847.509.9700 Fax 847.509.9798
www.ipc.org

The Principles of Standardization

In May 1995 the IPC's Technical Activities Executive Committee adopted Principles of Standardization as a guiding principle of IPC's standardization efforts.

Standards Should:
- Show relationship to Design for Manufacturability (DFM) and Design for the Environment (DFE)
- Minimize time to market
- Contain simple (simplified) language
- Just include spec information
- Focus on end product performance
- Include a feedback system on use and problems for future improvement

Standards Should Not:
- Inhibit innovation
- Increase time-to-market
- Keep people out
- Increase cycle time
- Tell you how to make something
- Contain anything that cannot be defended with data

Notice

IPC Standards and Publications are designed to serve the public interest through eliminating misunderstandings between manufacturers and purchasers, facilitating interchangeability and improvement of products, and assisting the purchaser in selecting and obtaining with minimum delay the proper product for his particular need. Existence of such Standards and Publications shall not in any respect preclude any member or nonmember of IPC from manufacturing or selling products not conforming to such Standards and Publication, nor shall the existence of such Standards and Publications preclude their voluntary use by those other than IPC members, whether the standard is to be used either domestically or internationally.

Recommended Standards and Publications are adopted by IPC without regard to whether their adoption may involve patents on articles, materials, or processes. By such action, IPC does not assume any liability to any patent owner, nor do they assume any obligation whatever to parties adopting the Recommended Standard or Publication. Users are also wholly responsible for protecting themselves against all claims of liabilities for patent infringement.

IPC Position Statement on Specification Revision Change

It is the position of IPC's Technical Activities Executive Committee (TAEC) that the use and implementation of IPC publications is voluntary and is part of a relationship entered into by customer and supplier. When an IPC standard/guideline is updated and a new revision is published, it is the opinion of the TAEC that the use of the new revision as part of an existing relationship is not automatic unless required by the contract. The TAEC recommends the use of the lastest revision. Adopted October 6. 1998

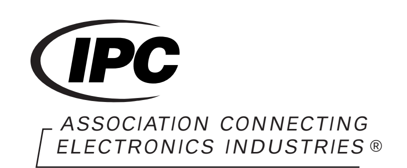
Why is there a charge for this standard?

Your purchase of this document contributes to the ongoing development of new and updated industry standards. Standards allow manufacturers, customers, and suppliers to understand one another better. Standards allow manufacturers greater efficiencies when they can set up their processes to meet industry standards, allowing them to offer their customers lower costs.

IPC spends hundreds of thousands of dollars annually to support IPC's volunteers in the standards development process. There are many rounds of drafts sent out for review and the committees spend hundreds of hours in review and development. IPC's staff attends and participates in committee activities, typesets and circulates document drafts, and follows all necessary procedures to qualify for ANSI approval.

IPC's membership dues have been kept low in order to allow as many companies as possible to participate. Therefore, the standards revenue is necessary to complement dues revenue. The price schedule offers a 50% discount to IPC members. If your company buys IPC standards, why not take advantage of this and the many other benefits of IPC membership as well? For more information on membership in IPC, please visit www.ipc.org or call 847/790-5372.

Thank you for your continued support.



IPC-2531

SMEMA Standard Recipe File Format Specification

Developed by the SMEMA Council of IPC

Users of this standard are encouraged to participate in the development of future revisions.

Contact:

IPC
2215 Sanders Road
Northbrook, Illinois
60062-6135
Tel 847 509.9700
Fax 847 509.9798

SMEMA

Standard Recipe File Format Specification

Version 1.0

DISCLAIMER

By publication of these standards, Surface Mount Equipment Manufacturers Association (SMEMA) takes no position respecting the validity of any patent rights or copyrights asserted in connection with any item mentioned in these standards. Under the current law of the United States and numerous other countries, the use of a copyright notice is not necessary in order for a work to have copyright protection. Such protection is available to all subject matter entitled to copyright protection from the moment it is "fixed in a tangible medium of expression." Therefore, unless the work has been published with a statement from the author that it may be freely reproduced, it should be assumed that all referenced works have copyright protection. Users of these standards are expressly advised that determination of any such patent rights or copyrights, and the risk of infringement of such rights are entirely their own responsibility.

These standards do not purport to address safety issues, if any, associated with their use. It is the responsibility of the user of these standards to establish appropriate safety and health practices and determine the applicability of regulatory limitations prior to use. SMEMA makes no warranties or representations as to the suitability of the standards set forth herein for any particular application. The determination of the suitability of the standard is solely the responsibility of the user. Users are cautioned to refer to manufacturer's instructions, product labels, product data sheets, and other relevant literature respecting any materials mentioned herein. These standards are subject to change without notice.

The user's attention is called to the possibility that compliance with this standard may require use of copyrighted material or of an invention covered by patent rights. By publication of this standard, SMEMA takes no position respecting the validity of any patent rights or copyrights asserted in connection with any item mentioned in this standard. Users of this standard are expressly advised that determination of any such patent rights or copyrights, and the risk of infringement of such rights, are entirely their own responsibility.

Participants

Task Force Leader
Andrew Dugenske, Georgia Institute of Technology

Contributors
Bob Balog, MPM
Dick Brown, 3Com
Steve Carlson, Research International
Loring Chadwick, MPM
Randy Chancellor, Mitron
Ranjan Chatterjee, Motorola
Norm Cox, Research International
Rob Donley, UNICAM
Ron Evans, 3Com
Stephen Fuks, Ford Motor Company
Kamran Guivian, Asymtek
Scott Hansohn, CyberOptics
Jeff Hawthorne, MV Technology
Matthew Kelly, Graphicode
Robert Kelley, General Scanning
Dave Kerem, Camelot Systems
Brian Leedy, Electrovert
Ken Moore, Georgia Institute of Technology
Thomas Newton, Panasonic KME
Eric Nillson, Graphicode
Curtis Parks, National Institute of Standards and Technology
Greg Parks, CyberOptics
Pete Patel, 3Com
Mark Pinkerton, Siemens
Scott Post, Delco Electronics
John Rosenberger, Universal Instruments
Steve Schwarz, Panasonic
Pat Sugrue, DEK Printing Machines
Shaun Turner, Fuji Machine Manufacturing
Steve Vickers, Zevatech
Chris Woodward, Siemens
Stefan Zühlke, Siemens

Table of Contents

1. Introduction

1.1. Intent

The intent of this specification is to provide a standard method for developing process control files used by electronics manufacturing equipment. Process control files (often referred to as recipes) provide the instruction sets used by assembly equipment to accomplish specified tasks.

In the past, proprietary file formats were the norm. By standardizing process control files, SMEMA's goal is to simplify the exchange of information on the factory floor by fostering interoperability. Through the use of this standard, it is believed significant cost savings and greater flexibility can be realized by software developers, equipment suppliers, and electronics manufacturers.

1.2. Scope

General

The purpose of this specification is to outline the requirements that an SRFF file must meet. The specification describes the file format, outlines the file sections, and indicates how data should be represented through objects. Objects can either be vendor independent (generic objects defined in this document) or vendor specific objects (objects created by a vendor). This document also includes error codes that should be used to report specific information about improperly constructed files. General guidelines for producing an SRFF file and vendor specific objects are also included.

Intended Audience

The intended audience for this document are individuals with knowledge of surface mount equipment, process control files, and the processes used to manufacture electronic products. Typical users might include manufacturing engineers, software tool developers, equipment operators, and application engineers.

1.3. Overview

This specification is divided into sections as listed below:

Section 1, Introduction

This section contains the scope and intent of this standard. A brief overview of each section is also included.

Section 2, General Guidelines

This section provides the general guidelines and specific requirements for developing an SRFF file. It also indicates the conformance requirements that a vendor must follow to be SRFF compliant.

Section 3, File Format

This section outlines the file type, delimitation and structure.

Section 4, Schema

This section indicates how the schema should be constructed and objects defined.

Section 5, Data

This section describes the method for representing data using the objects defined by the schema.

Section 6, Vendor Independent Objects

This section lists vendor independent objects by process type.

Section 7, Vendor Specific Objects

This section describes the method for constructing vendor specific objects. Vendor specific objects are used to augment the list of vendor independent objects. This standard will not attempt to catalog vendor specific objects. If a particular Specific object becomes widely used, it is anticipated that the object will become part of the independent object list.

Section 8, Error Codes

This section indicates how errors encountered in standard recipe files will be handled.

Section 9, Glossary

This section contains terms and definitions used in this standard.

Appendix A, Backus-Naur-Form Reference

Backus-Naur-Form (BNF) is used to define the syntax of an SRFF file. A BNF reference is included in this appendix.

Appendix B, BNF Grammar

The BNF grammar used to define the syntax of an SRFF file is included in this appendix.

Appendix C, Data Types

Definitions for the allowable SRFF file data types are included in this appendix.

Appendix D, Vendor Independent Object List

Appendix D contains a list of all the vendor independent objects that have been defined by this specification. The definitions include a description, sample schema entry, and a sample data entry. The definitions also include attribute names, data types, and descriptions. The objects are organized by process type.

Appendix E, Entity Relationship Diagram

An entity relationship diagram for all the vendor independent objects is included in this appendix.

Appendix F, Coordinate Systems

The coordinate system definitions are included in this appendix.

Appendix G, File Example

A sample SRFF file is included in this appendix.

Appendix H, Error Codes

Error codes that are to be returned by SRFF compliant software or equipment is included in this appendix.

Appendix I, Object Naming Form

If a vendor develops an object (vendor specific object), the object should be documented so that ambiguities do not arise. The form (or one similar) included in this appendix should be used for this purpose. If figures are required to document the object, they should be included with the form.

Appendix J, Compliance Forms

If software or equipment is indicated to be SRFF compliant, the forms (or similar) included in this appendix should be supplied with the product.

Appendix K, Method to Obtain Vendor Specific Object Tag from SMEMA

This appendix indicates the method for obtaining a vendor specific object tag from SMEMA.

2. General Guidelines

This section provides general guidelines for producing an SRFF file.

2.1. Priorities

When considering conflicting objectives for an SRFF file, the priority for developing a file should be as follows:

1. Facilitate automatic generation
2. Facilitate manual editing
3. Facilitate manual generation

2.2. Precedence

- Data contained in an SRFF file supersedes data external to the file.
- Vendor specific data supersedes vendor independent data.

2.3. Conformance Requirements

General

Compliance to the SRFF standard can be achieved at various levels, with one being the minimum level with the lowest performance. Compliance is indicated by the information provided by the vendor on the forms contained in Appendix J. A copy of these completed forms should be included in the documentation supplied with equipment or software that is indicated to be SRFF compliant.

Object Description Form

All vendor specific objects must be defined by the object naming form contained in Appendix I.

Vendor Specific Error Codes

For all software or equipment that is indicated to be SRFF compliant, the vendor must supply a list of error codes that might be returned by the software or equipment.

Vendor Specific Registration

To produce vendor specific objects, a vendor must register with SMEMA and obtain a vendor specific object tag. Appendix K contains the logistics for obtaining a vendor specific object tag from SMEMA.

2.4. Information Content

General

An SRFF file shall contain all the necessary data required by a single piece of process equipment to produce a product. (Note: Although a future goal of the SRFF specification is for all the data required to produce a product reside in a single file, only the process data that will be used by a single machine shall be contained in an SRFF file adhering to this specification.)

Units

Units of measure used in an SRFF file shall be defined once for the entire file. Data defining the units of measure must be located in the vendor independent data section.

Coordinate Systems

All coordinate systems shall adhere to the conventions shown in figure 2.1. Additional coordinate system conventions can be found in Appendix F.

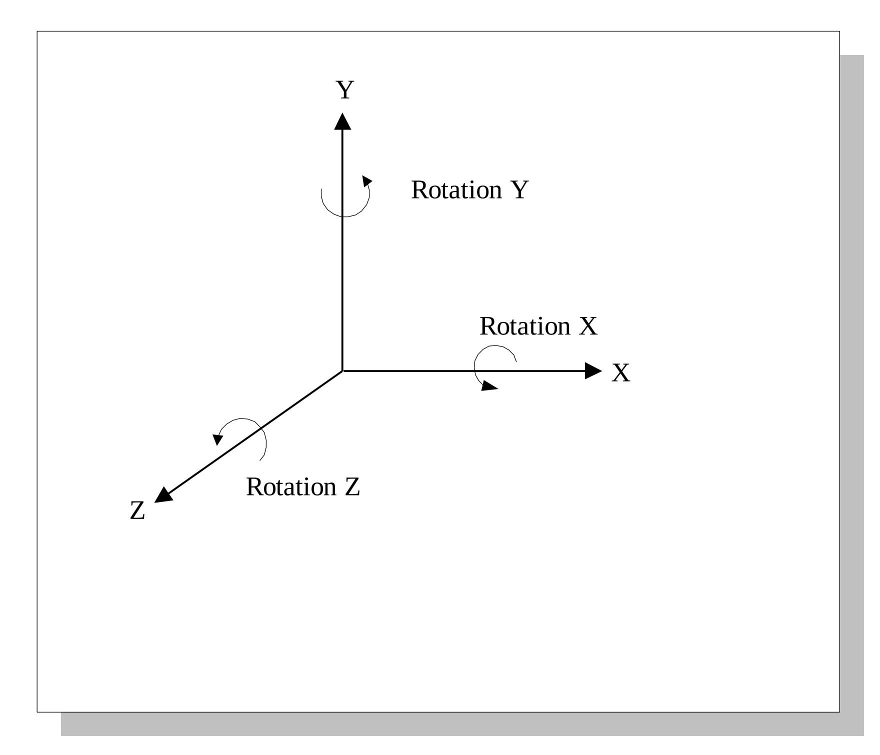


Figure 2.1 Coordinate system conventions to be used in SRFF files.

3. File Format

This section indicates how an SRFF file is structured and delimitated. It also indicates which characters can be used in an SRFF file and how comments are labeled.

3.1. Characters

An SRFF file can only contain ASCII characters. Binary data must be converted to ASCII and can only exist in the vendor specific data sections. ASCII and Binary are defined in Appendix C. The method to convert Binary data into ASCII data is also contained in Appendix C.

3.2. Comments

The # character is a comment character. All text on a line after the # character shall be considered comment text.

3.3. Delimitation

An SRFF file is delimited by position. All stand-alone groupings of letters or numbers must be delimited (separated) by one of the three combinations:

- White Space(s)
- [White Space(s)] { [White Space(s)]
- [White Space(s)] } [White Space(s)]

Where

- White Space(s) means one or more: space, tab, carriage return or line feed.
- [White Space(s)] means optional.
- The placement of { and } are dictated by the BNF grammar.

3.4. File Structure

An SRFF file contains two main sections: the schema and data. Each of these main sections is divided into product and process sections. The product and process sections are further divided into vendor independent and vendor specific sections.

The schema is placed at the beginning of the file, and defines the objects that will be used in the data section. The product section of the schema is used to define objects that pertain to the physical characteristics of the electronic product (e.g., location, thickness, and part numbers). Whereas, the process section of the schema is used to define objects that relate to the manufacturing of the product (e.g., placement order, squeegee pressure).

The schema can define two different types of objects: vendor independent and vendor specific. Vendor independent objects are defined by this standard and are meant to represent data and processes in a generic manner. Vendor specific objects are objects that have been defined by a particular vendor for a specific application. To foster interoperability, it is recommended that vendor independent objects be used whenever possible.

The data section of the file follows the schema. It contains instances of populated objects that were defined in the schema. As in the schema, the data section is segmented into product and process sections and each of these sections is segmented further into vendor independent and vendor specific sections.

```
{Schema
      {Product
            {Vendor Independent Product Schema
            {Vendor A Product Schema}
            {Vendor B Product Schema}
                        .
                        .
                        .
            {Vendor N Produce Schema}
      }
      {Process
            {Vendor Independent Process Schema}
            {Vendor A Process Schema}
            {Vendor B Process Schema}
                        .
                        .
                        .
            {Vendor N Process Schema}
      }
}
{Data
      {Product
            {Vendor Independent Product Data}
            {Vendor A Product Data}
            {Vendor B Product Data}
                        .
                        .
                        .
            {Vendor N Product Data}
      }
      {Process
            {Vendor Independent Process Data}
            {Vendor A Process Data}
            {Vendor B Process Data}
                        .
                        .
                        .
            {Vendor N Process Data}
      }
}
```

Figure 3.1 Pseudo code representation of an SRFF File.

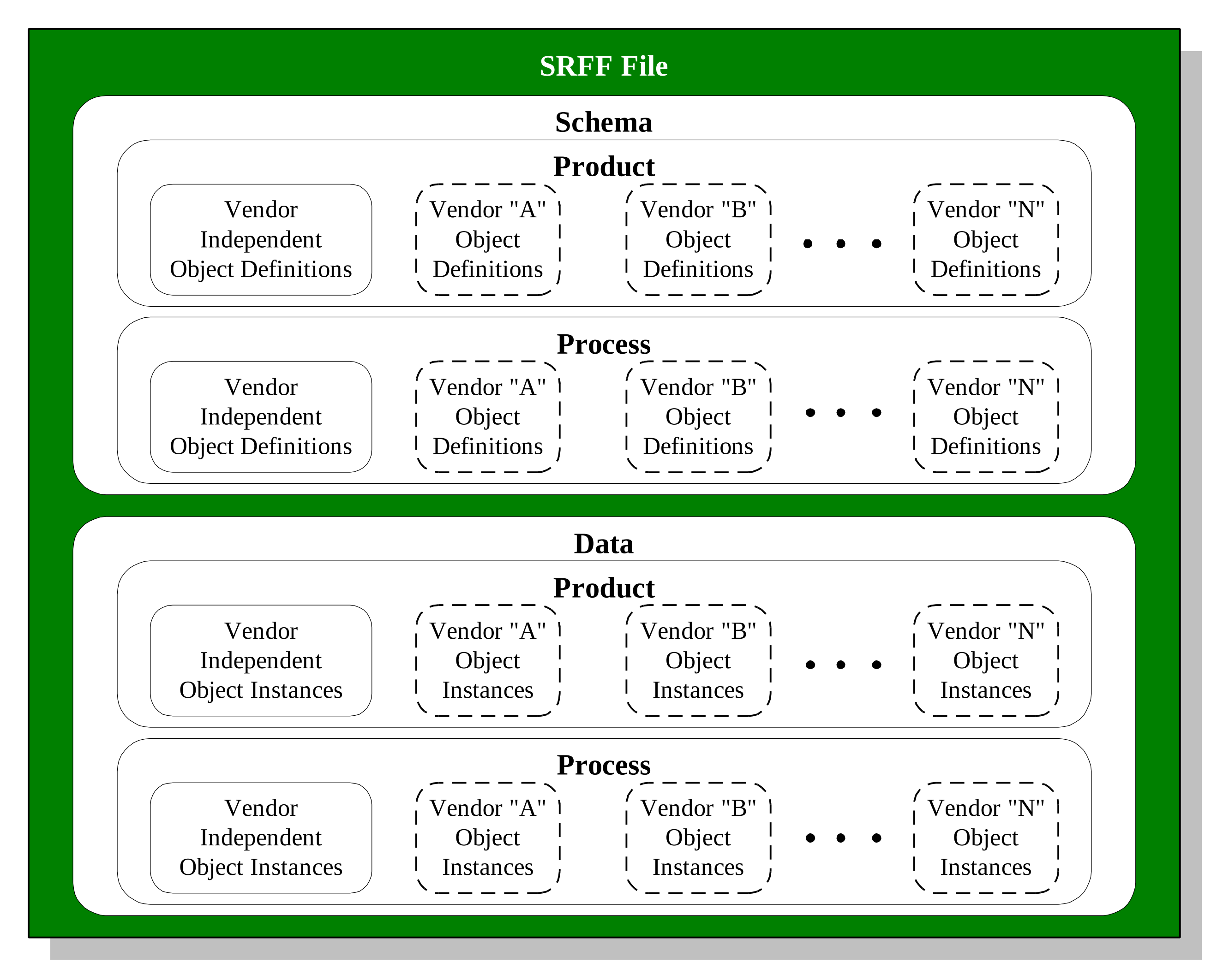


Figure 3.2 Graphical representation of an SRFF file.

4. Schema

4.1. General

The first section of the file is always a schema that provides instructions for parsing the file. The schema can define two types of objects: vendor independent and vendor specific.

4.2. Object Definition

Uniqueness

All objects used in the file must be defined by the schema and be unique. Objects can only be defined in terms of object and attribute identifiers that have been previously defined. Objects cannot be redefined.

Combining Objects

- Vendor independent objects may not be modified.
- Vendor specific objects may contain vendor independent objects.
- Vendor specific objects may not contain vendor specific objects defined by another vendor.
- Vendor specific objects may contain vendor specific objects defined by the same vendor.

Object attributes

Each object attribute has a data type assigned to it. Refer to Appendix C for the list of data types.

Schema Syntax

The BNF grammar for the schema is located in Appendix B. A BNF reference is contained in Appendix A.

5. Data

The Data section of the file contains product and process data in a format specified by the Schema.

5.1. Object Location

- Vendor independent product objects can appear in any product data section.
- Vendor independent process objects can appear in any process data section.
- Vendor specific product objects shall reside only in the corresponding vendor specific, product data section.
- Vendor specific process objects shall reside only in the corresponding vendor specific, process data section.

5.2. Object Order

The order of objects within a data section does not matter.

5.3. Attribute values

- Objects with the same name cannot have duplicate Id values.
- Default attribute values are not supported.
- Unused attribute values shall be indicated by the character "*".

5.4. Data Extension

- Data "carried over" from a previous object is not supported.
- Incremental values from previous objects are not supported.

6. Vendor Independent Objects

Vendor independent objects are maintained by SMEMA and provide a standard way to represent data that is commonly used in process control files. These objects do not include vendor or machine specific information, so they can easily be shared among platforms.

Developers of standard recipe files are encouraged to make use of these objects to improve compatibility. This section lists all the vendor independent objects by process type. A complete description of all the vendor independent objects is contained in Appendix D.

6.1. Common Objects

Table 6.1

Name	*Description*
Barcode	A product object defining the content of the product barcode on a panel.
ComponentDefinition	Used to link a component part name to a Location.
ComponentLink	Used to link a component package type to a part name.
Feature	Used to indicate a shape and the position and orientation of the shape with respect to a Pattern coordinate system.
FeatureGroup	A list of Pattern Features.
FeatureGroupOrdered	A list of Pattern Features where the order is significant.
Header	Used to include product notes and the product name in a predictable format.
Image	Used to define the position and orientation of an Image coordinate system with respect to a reference Image coordinate system. Associates an ImageDefinition and a SkipMark with the Image.
ImageDefinition	Used to group a set of Locations.
ImageFiducial	Associates a fiducial with an Image. Defines the shape, position, and orientation of the fiducial and associates a reference designator.
LocalFiducial	Associates a fiducial with a Location. Defines the shape, position, and orientation of the fiducial and associates a reference designator.
Location	Used to define the position and orientation of a component coordinate system with respect to an ImageDefinition coordinate system.
LocationGroup	A list of Locations and corresponding Images.
LocationGroupOrdered	A list of Locations and corresponding Images where the order is significant.
Panel	Used to define the dimensions of a Panel.
Pattern	Used to link a PatternDefinition to a part name.
PatternDefinition	Used to group a list of features.
Shape	Used to indicate a geometry type.
SkipMark	Used to define an Image SkipMark. Defines the shape, position, and orientation of the SkipMark.
SRFFVersion	Indicates the SMEMA SRFF version used to create the file.
VendorShapeLink	Used to link a vendor defined shape to the Shape object.

6.2. Dispense Objects

Table 6.2

Name	*Description*
DispenseOrder	A list of Pattern Features where the order is significant with respect to dispense operations.

6.3. Inspection Objects

Table 6.3

Name	*Description*
InspectOrder	A list of Pattern Features where the order is significant with respect to inspection operations.

6.4. Line Configuration Objects

As of this version of the specification, no vendor independent line configuration objects have been defined.

6.5. Material Movement Objects

As of this version of the specification, no vendor independent material movement objects have been defined.

6.6. Placement Objects

Table 6.4

Name	*Description*
PlacementOrder	A list of Locations where the order is significant with respect to placement operations.

6.7. Print Objects

Table 6.5

Name	*Description*
Print	A process object defining the Print stroke action parameters.
PrinterAlignment	A process object defining alignment fiducial information for a screen.
PrintArea	Product information defining the area to be printed.
ScreenProperties	A product object that defines the dimensions of a screen frame and the position of the Image.
ScreenFiducial	A process object defining a fiducial on a printer screen.
Squeegee	Process settings for a squeegee.
SqueegeeProperties	A process object containing information about squeegees.

6.8. Reflow Objects

As of this version of the specification, no vendor independent reflow objects have been defined.

6.9. Shape Object

Table 6.6

Name	*Description*
Cross	Used to define the two dimensional geometry of a cross shape.
Diamond	Used to define the two dimensional geometry of a diamond shape.
Disc	Used to define the two dimensional geometry of a disc shape.
Donut	Used to define the two dimensional geometry of a donut shape.
Rectangle	Used to define the two dimensional geometry of a rectangle shape.
Triangle	Used to define the two dimensional geometry of a triangle shape.

6.10. Test Objects

As of this version of the specification, no vendor independent test objects have been defined.

6.11. Unit Objects

Table 6.7

Name	*Description*
AcclerationUnits	Used to define the units of acceleration.
AngleUnits	Used to define the units of angular measurement.
AngularAcclerationUnits	Used to define the units of angular acceleration.
AngularVelocityUnits	Used to define the units of angular velocity.
DistanceUnits	Used to define the units of distance.
FlowUnits	Used to define the units of volumetric flow.
ForceUnits	Used to define the units of force.
HumidityUnits	Used to define the units of humidity.
MassUnits	Used to define the units of mass.
PowerUnits	Used to define the units of power.
PressureUnits	Used to define the units of pressure.
TemperatureUnits	Used to define the units of temperature.
TimeUnits	Used to define the units of time.
TorqueUnits	Used to define the units of torque.
VelocityUnits	Used to define the units of velocity.
VolumeUnits	Used to define the units of volume.

6.12. Wave Solder Objects

As of this version of the specification, no vendor independent wave solder objects have been defined.

7. Vendor Specific Objects

This standard allows vendors to develop vendor specific objects that can be used in an SRFF file. A vendor must first register with SMEMA and obtain a vendor specific object tag prior to including vendor specific objects in an SRFF file. For information on registering with SMEMA as an SRFF vendor, please see Appendix K.

As certain vendor specific objects become widely used it is anticipated that SMEMA will incorporate them into future versions of this specification as vendor independent objects. By incorporating new vendor independent objects into future versions of this specification interoperability should be improved.

Once a vendor has obtained an object tag from SMEMA, vendor specific object definitions can be included in the appropriate vendor specific object section of an SRFF file. A vendor specific section starts with the keyword “Organization”, and the appropriate location in the file is dictated by the BNF grammar contained in Appendix B. Rules and guidelines for developing a vendor specific object are as follows:

The following are Requirements for Producing Vendor Independent Objects:

1. The BNF grammar must be adhered to. See Appendix B.
2. The rules outlined in Section 4 (Schema) must be followed.
3. All vendor specific objects must be documented using a form contained in Appendix I (or one similar).
4. A vendor specific object name must start with a vendor specific object tag assigned by SMEMA. Appendix K indicates the method for obtaining a vendor specific object tag (i.e. DugNewObjectName, where Dug is the vendor tag).
5. Object names must be unique.
6. Attribute names within an object must be unique.
7. Object and attribute names must be less than 64 characters and start with a letter.
8. All objects must contain an Id.
9. Objects must be designated as product or process type.

The following are Guidelines for Producing Vendor Independent Objects:

1. Development of new unit objects is discouraged.
2. Product objects should not reference process objects.
3. Hungarian Notation should be used to specifying object and attribute names (i.e. AddNewObjectName should be used versus addNewobjname).
4. Underscores should be avoided in object and attribute names.
5. A minimum use of abbreviations should be used when naming objects and attributes (i.e. AddNewObjectName should be used versus AddNwObjNam).
6. Words that should be avoided included: left, right, up, down, and front. Instead, PositionX, PositionY, and PositionZ should be used and referenced to an image.
7. It is recommended that the conventions outlined in table 7.1 be used for defining vendor specific object attributes.

Table 7.1

Attribute Name[1]	**Type**	**Order**[2]	**Description**
*Objectname*ID	Id	1	Object Id
Name	String	2	Name associated with object instance
PositionX	Distance	3	Distance in x from the reference image origin
PositionY	Distance	4	Distance in y from the reference image origin
PositionZ	Distance	5	Distance in z from the reference image origin
RotationX	Angle	6	The Rotation about the reference image x-axis.
RotationY	Angle	7	The Rotation about the reference image y-axis.
RotationZ	Angle	8	The Rotation about the reference image z-axis.
RefernceImageId	Id	Last -2	Id of the reference image.
Reference*Name1*	Id	Last -1	Reference to the Id of Object Name1
Reference*Name2*	Id	Last	Reference to the Id of Object Name2

[1]Italics indicate user definable.
[2]Order indicates the relative position of the attribute in the object definition.

8. Error Types

Errors produced by SRFF compliant software or equipment should be classified as lexical, syntactical, semantic, or dynamic by the software or equipment processing a file. The four classifications are defined below.

Lexical

These errors are found when the compiler is parsing the stream of characters constituting the source file into a sequence of language tokens, which in turn will form statements or expressions. An example of a lexical error could be “garbage” end-of-file control characters appearing after the final right brace of the SRFF program. Lexical type errors also occur from the misspelling of an identifier, keyword or operator.

Syntax

Syntactical errors are errors with the format of a particular statement or expression which violate the rules of the SRFF grammar. These errors can occur when a token of an expression is either missing, contains an extraneous character, was replaced by an invalid character, or was transposed to an incorrect location within the expression. Examples of a syntax error include:

Example	*Error Code*	*Error*
{SRFFVersion 1 “1.0”	SRFF 3008	Right Bracket Missing
{SRFFVersion 1 “”1.0”}	SRFF 3010	Extraneous Character
[SRFFVersion 1 “1.0”}	SRFF 3007	Left Bracket Missing
{1 SRFFVersion “1.0”}	SRFF 3001	Invalid Object Name

Semantic

Semantic errors occur when the expression has the correct syntactic structure but has no meaning to the operation involved. A generic example of a semantic type error would be if an expression attempted to add two identifiers, one of which is the name of an array, and the other the name of a procedure. Primary sources of semantic errors are undeclared/misdeclared names and data type incompatibilities. SRFF related examples of semantic type errors would be the use of a component identifier in the list of fiducials identifiers used for image translation correction. Also, the use of the characters ABC for the X location coordinate of a component placement would constitute a semantic error.

Dynamic

Dynamic errors do not necessarily relate to the SRFF language but deal with errors which happen during the processing of the source program. These errors are usually detected during run time. Examples of dynamic errors include:

- The system could not allocate sufficient memory to complete the SRFF file import.
- The local component database to be updated could not be found.
- The placement location is off the panel.

8.1. Error Codes

Appendix H contains a standard set of error codes that shall be returned by SRFF compliant equipment and software for syntactical and lexical errors. Semantic and dynamic error codes are beyond the scope of this document.

8.2. Mechanism for Reporting Error Messages

No mechanism for reporting errors is explicitly part of this specification.

9. Glossary

Backus-Naur Form (BNF). The language used to define the structure and syntax of an SRFF file and the objects used in an SRFF file.

Delimitation. The method of separating individual groupings of letters or numbers. Delimiters mark the beginning and end of a grouping

Feature. A shape that has a position and orientation with respect to a pattern origin.

Image. A coordinate system definition.

Image Definition. A group of locations.

Location. The position of a component.

Object Definition. A definition that includes the object name, object structure, attributes names, and attribute types (similar to the table definition in a database).

Object Instance. Data contained in the structure defined by an object (similar to individual records in a database table).

Objects. A structure used to represent information (similar to a database table definition). In this specification, object methods are not supported.

Panel. The substrate that will be passed through the manufacturing equipment. All the coordinate systems in an SRFF file reference the panel coordinate system directly or indirectly.

Pattern. A geometric arrangement of shapes associated with a component.

Process. The section of a file pertaining to the manufacturing process of the electronic product, such as the inspection or placement order.

Product. The section of a file pertaining to the physical characteristics of the electronic product, such as dimensions and location.

Schema. The section of a file used to provide object definitions.

Vendor Independent Object. An object defined by this standard for generic data and processes.

Vendor Specific Object. An object defined by a vendor for use in specific applications.

Appendix A
Backus-Naur-Form Reference

General

We use the Backus-Naur-Form (BNF) to define the syntax of an SRFF file. Every BNF definition follows the form:

Symbol :- Rule

Where, *symbol* is the name of the symbol to be defined and *rule* is the direction that leads to the symbol.

Concatenation

A concatenation of symbols is described by combining single symbols. As an example, the definition:

SmemaFile :- SchemaSection DataSection

means that a SmemaFile consists of a SchemaSection followed by a DataSection

Multiple Choices

The operator '|' is used to select between rules. The '|' can separate more than one rule. As an example:

Digit :- 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9

means that a digit is 0 or 1 or ... or 9.

Example (Defining an Integer)

An Integer is either a digit (i.e. 2) or several digits (i.e. 4711). The BNF-Notation might be as follows:

```
Integer   :-  Digit               // a single digit
           |  Digit Integer       // repetition (recursive)
```

Example (Defining a String)

The notation for a string enclosed in quotation marks might be:

```
String    :- '"' meat '"'
Meat      :-  EMPTY          // allows empty strings
           |  Letter Meat
Letter    :- 'a' | 'b' | 'c' | .......
```

Appendix B
BNF Grammar

```
Angle                  :-  (A|a)(N|n)(G|g)(L|l)(E|e)
AngularAcceleration    :-  (A|a)(N|n)(G|g)(U|u)(L|l)(R|r)(A|a)(C|c)(C|c)(E|e)(L|l)(E|e)(R|r)(A|a)(T|t)(I|i)
                           (O|o)(N|n)
AngularVelocity        :-  (A|a)(N|n)(G|g)(U|u)(L|l)(R|r)(V|v)(E|e)(L|l)(O|o)(C|c)(I|i)(T|t)(Y|y)
Binary                 :-  (B|b)(I|i)(N|n)(A|a)(R|a)(Y|y)
Boolean                :-  (B|b)(O|o)(O|o)(L|l)
Close                  :-  }
Data                   :-  (D|d)(A|a)(T|t)(A|a)
DateTime               :-  (D|d)(A|a)(T|t)(E|e)(T|t)(I|i)(M|m)(E|e)
Distance               :-  (D|d)(I|i)(S|s)(T|t)(A|a)(N|n)(C|c)(E|e)
Dot                    :-  .
Empty                  :-
False                  :-  (F|f)(A|a)(L|l)(S|s)(E|e)
Float                  :-  (F|f)(L|l)(O|o)(A|a)(T|t)
Flow                   :-  (F|f)(L|l)(O|o)(W|w)
Force                  :-  (F|f)(O|o)(R|r)(C|c)(E|e)
HexCode                :-  0x|0X
Humidity               :-  (H|h)(U|u)(M|m)(I|i)(D|d)(I|i)(T|t)(Y|y)
Id                     :-  (I|i)(D|d)
Integer                :-  (I|i)(N|n)(T|t)
Length                 :-  (L|l)(E|e)(N|n)(G|g)(T|t)(H|h)
List                   :-  (L|l)(I|i)(S|s)(T|t)
Mass                   :-  (M|m)(A|a)(S|s)(S|s)
Object                 :-  (O|o)(B|b)(J|j)(E|e)(C|c)(T|t)
OctalCode              :-  (0)
Open                   :-  {
Organization           :-  (O|o)(R|r)(G|g)(A|a)(N|n)(I|i)(Z|z)(A|a)(T|t)(I|i)(O|o)(N|n)
Power                  :-  (P|p)(O|o)(W|w)(E|e)(R|r)
Pressure               :-  (P|p)(R|r)(E|e)(S|s)(S|s)(U|u)(R|r)(E|e)
Process                :-  (P|p)(R|r)(O|o)(C|c)(E|e)(S|s)(S|s)
Product                :-  (P|p)(R|r)(O|o)(D|d)(U|u)(C|c)(T|t)
Schema                 :-  (S|s)(C|c)(H|h)(E|e)(M|m)(A|a)
Select                 :-  (S|s)(E|e)(L|l)(E|e)(C|c](T|t)
SMEMA                  :-  (S|s)(M|m)(E|e)(M|m)(A|a)
String                 :-  (S|s)(T|t)(R|r)(I|i)(N|n)(G|g)
StringDelimiter        :-  "
Temperature            :-  (T|t)(E|e)(M|m)(E|e)(R|r)(A|a)(T|t)(U|u)(R|r)(E|e)
Time                   :-  (T|t)(I|i)(M|m)(E|e)
Torque                 :-  (T|t)(O|o)(R|r)(Q|q)(U|u)(E|e)
True                   :-  (T|t)(R|r)(U|u)(E|e)
Unused                 :-  *
Velocity               :-  (V|v)(E|e)(L|l)(O|o)(C|c)(I|i)(T|t)(Y|y)
Volume                 :-  (V|v)(O|o)(L|l)(U|u)(M|m)(E|e)

SMEMAFile              :-  SchemaSection DataSection

SchemaSection          :-  Open Schema ProductSchema ProcessSchema Close
```

ProductSchema	:-	Open Product SMEMASchema VendorSchemas Close
ProcessSchema	:-	Open Process SMEMASchema VendorSchemas Close
SMEMASchema	:-	Open Organization SMEMA SchemaEntries Close
VendorSchemas	:-	VendorSchema
	|	VendorSchema VendorSchemas
	|	Empty
VendorSchema	:-	Open Organization VendorName SchemaEntries Close
VendorName	:-	Name
SchemaEntries	:-	SchemaEntry
	|	SchemaEntry SchemaEntries
	|	Empty
SchemaEntry	:-	Open ObjectName ObjectId Atttributes Close
ObjectName	:-	Name
ObjectId	:-	Open Id ObjectIdName Close
ObjectIdName	:-	Name
Attributes	:-	Attribute
	|	Attribute Attributes
	|	Empty
Attribute	:-	Open AttributeType AttributeName Close
	|	Open Select Open Attributes Close Close
	|	Open List Open Attributes Close Close
	|	ObjectAttribute
AttributeType	:-	Integer | Float | Boolean | Id | String | Binary | DateTime | Length | Angle | Distance | AngularAcceleration | AngularVelocity | Flow | Force | Humidity | Mass | Power | Pressure | Temperature | Time | Torque | Velocity | Volume
AttributeName	:-	Name
ObjectAttribute	:-	Open Object ObjectName Close

```
DataSection        :-  Open Data ProductData ProcessData Close

ProductData        :-  Open Product DataBlocks Close

ProcessData        :-  Open Process DataBlocks Close

DataBlocks         :-  Open Organization VendorName DataEntries Close
                   |   Open Organization VendorName DataEntries Close DataBlocks
                   |   Empty

DataEntries        :-  DataEntry
                   |   DataEntry DataEntries
                   |   Empty

DataEntry          :-  Open ObjectName BaseTenNumber Values Close

Values             :-  Value
                   |   Value Values
                   |   Empty

Value              :-  Unused
                   |   Open Values Close
                   |   DataEntry
                   |   BaseTenNumber
                   |   BaseOctalNumber
                   |   BaseHexNumber
                   |   IEEENumber
                   |   StringValue
                   |   BooleanValue
                   |   BinaryValue

BooleanValue       :-  True | False | ‘T’ | ‘F’ | ‘t’ | ‘f’

BinaryValue        :-  StringValue (note: has to follow uuencode rules)

BaseTenNumber      :-  Sign Digits

Digits             :-  Digit | Digit Digits

BaseHexNumber      :-  HexCode HexNumber

HexNumber          :-  HexDigit | HexDigit HexNumber

BaseOctalNumber    :-  OctalCode OctalNumber

OctalNumber        :-  OctalDigit | OctalDigit OctalNumber
```

```
IEEENumber          :-  BaseTenNumber Dot Digits (E|e) BaseTenNumber
                    |   BaseTenNumber (E|e) BaseTenNumber
                    |   BaseTenNumber Dot Digits
                    |   BaseTenNumber

Sign                :-  Empty | - | +

OctalDigit          :-  0 | 1 | 2 | 3 | 4 | 5 | 6 | 7

Digit               :-  0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9

HexDigit            :-  0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | (A|a) | (B|b) | (C|c) | (D|d) | (E|e) | (F|f)

StringValue         :-  StringDelimiter Letters StringDelimiter

Letters             :-  Empty | Character Letters

Letter              :-  a | b | .... | z | A | B | .... | Z

SpecialCharacter    :-  _ | § | $ | % | & | / | = | * | + | - | # | \ | ? | ! | . | , | ; | : | < | > | @ | ^ | °

Character           :-  Letter | Digit | SpecialCharacter

Name                :-  NameCharacters
                    |   Empty

NameCharacter       :-  Letter
                    |   Digit
                    |   Empty

NameCharacters      :-  NameCharacter
                    |   NameCharacter NameCharacters
                    |   Empty
```

Appendix C
Data Types

Integer data type definition
Both positive and negative integers are allowed without requiring or specifying a range of valid values.

Floating point data type definition
Floating point numbers are allowed which meet the IEEE-754 specification for binary floating point data.

Boolean data type definition
Boolean data values are represented as specified in the BNF Grammar.

ID data type definition
ID values are integers, which are unique when required to specify a particular object of a particular type.

String data type definition
String data is defined as starting and ending with a quotation mark ("). All valid ASCII characters are permitted. Embedded quotation marks are preceded with a backslash (\). Embedded backslashes are also preceded with a backslash. (i.e. The string representing the filename of a directory d:\"fred"\ would be specified as "d:\\\"fred\"\\"). String data is limited to 65535 bytes per item. Newline characters are permitted within strings.

Binary data type definition
Binary data is allowed and should be encoded according to the uuencode algorithm. The data should start with the keyword "begin" followed by a line break. The first character of each line represents the number of bytes on the line, zero is represented with a back-quote or grave accent mark (`). A number is formed as a character with the space character (' ') added to it in order to assure that it is printable. A maximum of forty-five bytes can be on a single line. Three bytes of 8-bit data are then split into four bytes of 6-bit data, which are all shifted into the range of printable characters, and this process is repeated for the entire line, as needed for the whole file. See also
http://www.delorie.com/gnu/docs/sharutils/uuencode.5.html

DateTime data type definition
To specific a particular date and time, the DateTime type should be used. The DateTime type makes use of a string to represent data. The format of the string is as follows: YYYY-MM-DDThh:mm:ss.sTZD where,

YYYY = four-digit year
MM = two-digit month (01=January, etc.)
DD = two-digit day of month (01 through 31)
hh = two digits of hour (00 through 23)
mm = two digits of minute (00 through 59)
ss = two digits of second (00 through 59)
s = one or more digits representing a decimal fraction of a second
TZD = time zone designator (Z or +hh:mm or -hh:mm)

Note that the "T" appears literally in the string. 1963-03-25T08:14:34.56-05:00 is an example.
See also http://www.w3.org/TR/NOTE-datetime.html

Measurement data type definitions

In addition to dimensionless attribute types, the BNF grammar also defines several attribute types for which there are associated units. The values of these objects are float, the corresponding units are defined by a unit object instance contained in the SRFF. The following table shows the correspondence between the attribute type names and the respective unit object.

Attribute Type	**Units Object Name**
Acceleration	AccelerationUnits
Angle	AngleUnits
AngularAcceleration	AccelerationUnits
AngularVelocity	AngularVelocityUnits
Distance	DistanceUnits
Flow	FlowUnits
Force	ForceUnits
Humidity	HumidityUnits
Mass	MassUnits
Power	PowerUnits
Pressure	PressureUnits
Temperature	TemperatureUnits
Time	TimeUnits
Torque	TorqueUnits
Velocity	VelocityUnits
Volume	VolumeUnits

Table C.1 Unit objects used to define the unit values for a unit attribute type.

Appendix D
Object List

Barcode

Description Used to define the content of a barcode. The barcode may pertain to a product or component.

Schema Entry

```
{Barcode
{Id BarcodeId}
{String Barcode}
{Id ComponentLinkId}
}
```

Data Example

```
{Barcode 12 "VIN0412345" 33}
```

Attributes

Name	*Type*	*Description*
BarcodeId	Id	A unique number for each instance of the Barcode object.
Barcode	String	A string containing barcode characters.
ReferenceComponentLinkId	Id	The Id number of the corresponding ComponentLink object.

ComponentDefinition

Common Group
Product Category

Description Used to link a component part name to a Location.

Schema Entry

```
{ComponentDefinition
{Id ComponentDefinitionId}
{String PartName}
{Id ReferenceComponentLinkId}
}
```

Data Example

```
{ComponentDefinition 130 "SOT-3" 120}
```

Attributes

Name	***Type***	***Description***
ComponentDefinitionId	Id	A unique number for each instance of the ComponentDefinition object.
PartName	String	The component part name.
ReferenceComponentLinkId	Id	The Id number of the corresponding ComponentLink object.

ComponentLink

Common Group
Product Category

Description Used to link a component package type to a part name.

Schema Entry

```
{ComponentLink
{Id ComponentLinkId}
{String PackageName}
}
```

Data Example

```
{ComponentLink 120 "Quad Flat Pack 256"}
```

Attributes

Name	***Type***	***Description***
ComponentLinkId	Id	A unique number for each instance of the ComponentLink object.
PackageName	String	The name of the package type.

Feature

Description Used to define the shape, orientation and position of a pattern item.

Schema Entry

```
{Feature
{Id FeatureId}
{String FeatureName}
{Distance PositionX}
{Distance PositionY}
{Distance PositionZ}
{Angle RotationZ}
{Id ReferenceShapeId}
}
```

Data Example

```
{Feature 101 "Fist Pad" 230 230 0 180 103}
```

Attributes

Name	***Type***	***Description***
FeatureId	Id	A unique number for each instance of the Feature object.
FeatureName	String	The name of the feature.
PositionX	Distance	The distance in X from the pattern origin to this feature origin with respect to the Pattern coordinate system.
PositionY	Distance	The distance in Y from the pattern origin to this feature origin with respect to the Pattern coordinate system.
PositionZ	Distance	The distance in Z from the pattern origin to this feature origin, with respect to the Pattern coordinate system.
RotationZ	Angle	The rotation of this feature coordinate system with respect to the Z-axis of the Pattern coordinate system.
ReferenceShapeId	Id	The Id number of the Shape used for this feature.

FeatureGroup

Common Group
Process Category

Description A list of Pattern features.

Schema Entry

{FeatureGroup
{Id FeatureGroupId}
{LIST {
{Id ReferenceImageId}
{Id ReferenceLocationId}
{Id ReferencePatternDefinitionId}
{Id ReferenceFeatureId}
}}}

Data Example

{FeatureGroup 140
{
2 3 12 22
2 3 12 23
2 3 12 24
2 3 12 25
}}

Attributes

Name	*Type*	*Description*
FeatureGroupId	Id	A unique number for each instance of the FeatureGroup object.
ReferenceImageId	Id	The Id number of the corresponding Image.
ReferenceLocationId	Id	The Id number of the corresponding Location.
ReferencePatternDefinitionId	Id	The Id number of the corresponding Pattern.
ReferenceFeatureId	Id	The Id number of the corresponding Feature. The Id number is defined in a PatternDefinition object.

Notes

1. If a ReferencePatternDefinitionId and a ReferenceFeatureId are present in a list item, then the ReferencePatternId and the ReferencePatternDefinitionId must be contained in the same PatternDefinition.
2. An entire Pattern can be reference by setting the ReferenceFeatureId equal to *.

FeatureGroupOrdered

Common Group
Process Category

Description A list of Pattern features where the order is significant.

Schema Entry

```
{FeatureGroupOrdered
{Id FeatureGroupOrderedId}
{LIST {
{Id ReferenceImageId}
{Id ReferenceLocationId}
{Id ReferencePatternDefinitionId}
{Id ReferenceFeatureId}
}}}
```

Data Example

```
{FeatureGroupOrdered 1
{
4 3 12 22
4 3 12 23
4 3 12 24
4 3 12 25
}}
```

Attributes

Name	***Type***	***Description***
FeatureGroupOrderedId	Id	A unique number for each instance of the FeatureGroupOrdered object.
ReferenceImageId	Id	The Id number of the corresponding Image.
ReferenceLocationId	Id	The Id number of the corresponding Location.
ReferencePatternDefinitionId	Id	The Id number of the corresponding Pattern.
ReferenceFeatureId	Id	The Id number of the corresponding Feature. The Id number is defined in a PatternDefinition object.

Notes

1. If a ReferencePatternDefinitionId and a ReferenceFeatureId are present in a list item, then the ReferencePatternId and the ReferencePatternDefinitionId must be contained in the same PatternDefinition.
2. An entire Pattern can be reference by setting the ReferenceFeatureId equal to *.

Header

Description Used to include product notes and the product name.

Schema Entry

```
{Header
{Id HeaderId}
{String ProductName}
{String ProductNotes}
{DateTime Created}
{DateTime LastModified}
}
```

Data Example

```
{Header 1
{
"        Telephone Prototype        "

"      New Product Introduction
        for the 1998 Season         "

"1995-05-07T14:13:30.45-04:00"
"1996-02-12T13:45:23.34-04:00"
}
```

Attributes

Name	***Type***	***Description***
HeaderId	Id	A unique number for each instance of the Header object.
ProductName	String	The name of the product that the file defines.
ProductNotes	String	General notes about the file.
Created	DateTime	Date and time file was initially created.
LastModified	DateTime	Date and time file was last modified.

Notes

1. One header object instance must be included in each file.
2. Only one header object instance is allowed per file.

Image

Common Group
Product Category

Description

Used to define the position and orientation of an Image coordinate system with respect to a reference Image coordinate system. Associates an ImageDefinition and a SkipMark with the Image.

Schema Entry

{Image
{Id ImageId}
{Distance PositionX}
{Distance PositionY}
{Distance PositionZ}
{Angle RotationX}
{Angle RotationY}
{Angle RotationZ}
{Id ReferenceImageDefinitionId}
{Id ReferenceImageId}
{Id ReferenceSkipMarkId}
}

Data Example

{Image 170 16789 14758 * * * 24.5 1566 156 189}

Attributes

Name	*Type*	*Description*
ImageId	Id	A unique number for each instance of the Image object.
PositionX	Distance	The distance in X from the reference Image origin to this Image origin, with respect to the reference Image coordinate system.
PositionY	Distance	The distance in Y from the reference Image origin to this Image origin, with respect to the reference Image coordinate system.
PositionZ	Distance	The distance in Z from the reference Image origin to this Image origin, with respect to the reference Image coordinate system.
RotationX	Angle	The rotation of this Image coordinate system with respect to the X-axis of the reference Image coordinate system.
RotationY	Angle	The rotation of this Image coordinate system with respect to the Y-axis of the reference Image coordinate system.
RotationZ	Angle	The rotation of this Image coordinate system with respect to the Z-axis of the reference Image coordinate system.
ReferenceImageDefinitionId	Id	The Id number of the corresponding ImageDefinition for this Image.
ReferenceImageId	Id	The Id number of the reference Image.
ReferenceSkipMarkId	Id	The Id number of the corresponding SkipMark for this Image.

Notes

1. When mapping from the reference Image coordinate system to this Image coordinate system, the order of rotation is X, Y, then Z.
2. When the ImageId = 0, the Image refers to the Panel.

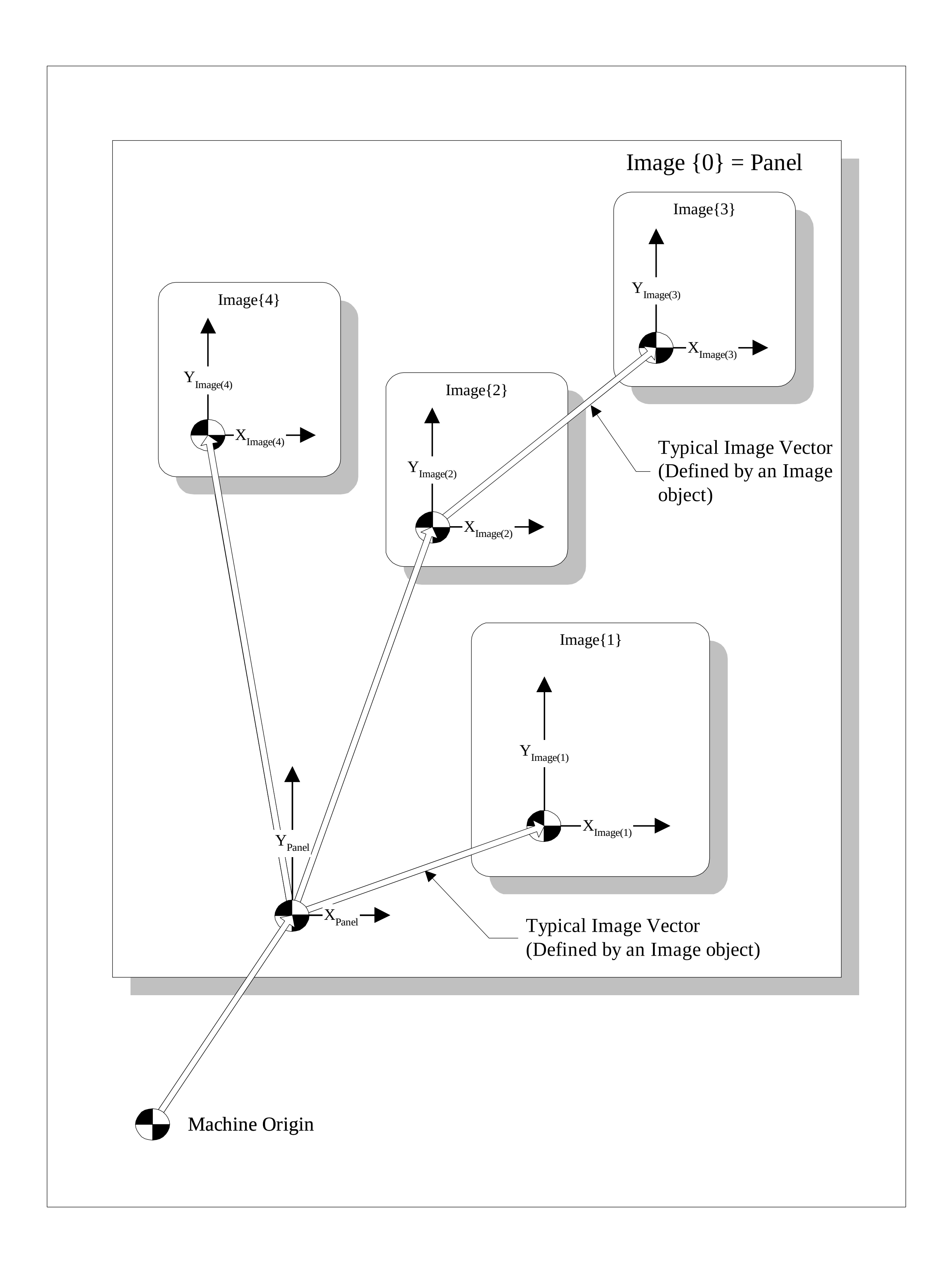


Figure D.1 A graphical representation of a machine coordinate system, a panel coordinate system, and various image coordinate systems.

ImageDefinition

Common Group
Product Category

Description Used to group a set of Locations.

Schema Entry

{ImageDefinition
{Id ImageDefinitionId}
{String ImageDefinitionName}
}

Data Example

{ImageDefinition 180 “Radio”}

Attributes

Name	***Type***	***Description***
ImageDefinitionId	Id	A unique number for each instance of the ImageDefinition object.
ImageDefinitionName	String	The name of the ImageDefinition.

Notes

1. The position and orientation of an ImageDefinition coordinate system are defined by an Image coordinate system.
2. When the ImageDefinitionId = 0, the ImageDefinition refers to the panel.

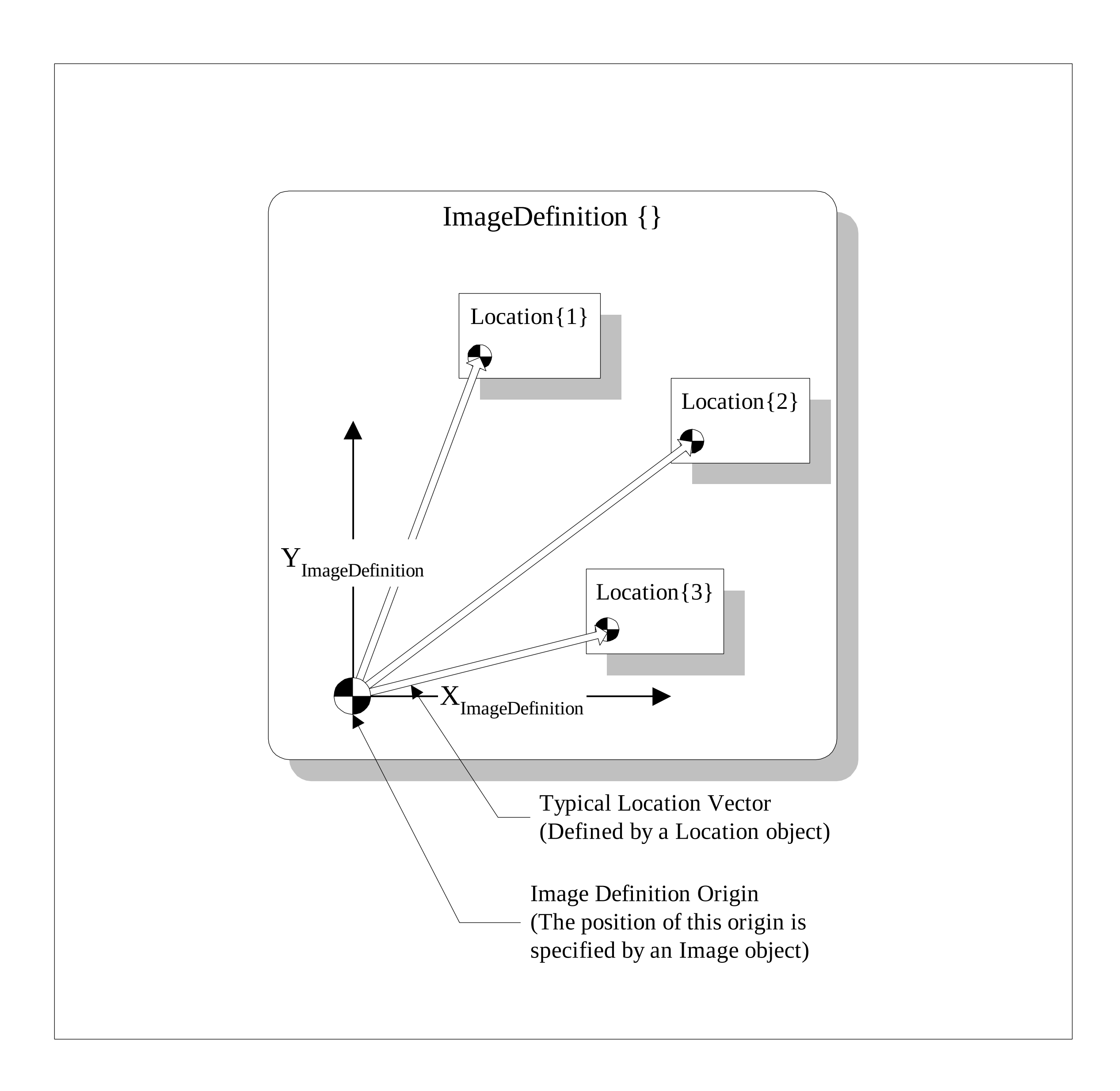


Figure D.2 An ImageDefinition with corresponding Locations.

ImageFiducial

Description Associates a fiducial with an ImageDefinition. Defines the shape, position, and orientation of the fiducial and associates a reference designator.

Schema Entry

```
{ImageFiducial
{Id ImageFiducialId}
{String ReferenceDesignator}
{Distance PositionX}
{Distance PositionY}
{Distance PositionZ}
{Angle RotationZ}
{Id ReferenceImageDefinitionId}
{Id ReferenceShapeId}
}
```

Data Example

{ImageFiducial 190 “IF1” 200 100 * 0 101 201}

Attributes

Name	*Type*	*Description*
ImageFiducialId	Id	A unique number for each instance of the ImageFiducial object.
ReferenceDesignator	String	The reference designator for this fiducial.
PositionX	Distance	The distance in X from the ImageDefinition origin to this fiducial origin, with respect to the ImageDefinition coordinate system.
PositionY	Distance	The distance in Y from the ImageDefinition origin to this fiducial origin, with respect to the ImageDefinition coordinate system.
PositionZ	Distance	The distance in Z from the ImageDefinition origin to this fiducial origin, with respect to the ImageDefinition coordinate system.
RotationZ	Angle	The rotation of the fiducial coordinate system with respect to the Z-axis of the ImageDefinition coordinate system.
ReferenceImageDefinitionId	Id	The Id number of the corresponding ImageDefinition for this fiducial.
ReferenceShapeId	Id	The Id number of the Shape used for this fiducial. The fiducial coordinate system defines the position and orientation of the shape coordinate system.

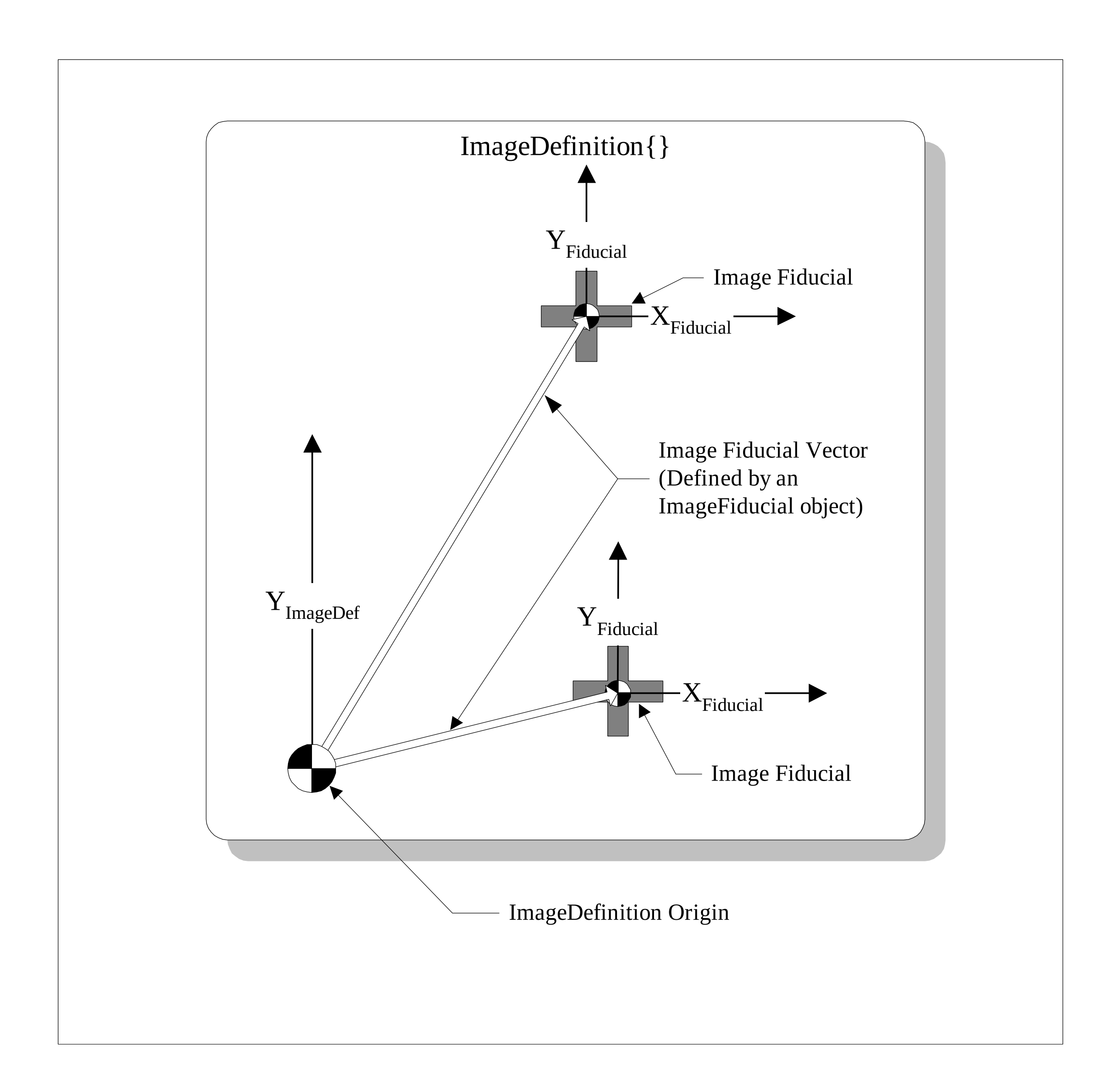


Figure D.3 An ImageDefinition with corresponding ImageFiducials.

LocalFiducial

Common Group
Product Category

Description Associates a fiducial with a Location. Defines the shape, position, and orientation of the fiducial and associates a reference designator.

Schema Entry

```
{LocalFiducial
{Id LocalFiducialId}
{String ReferenceDesignator}
{Distance PositionX}
{Distance PositionY}
{Distance PositionZ}
{Angle RotationZ}
{Id ReferenceLocationId}
{Id ReferenceShapeId}
}
```

Data Example

```
{LocalFiducial 200 “LF1” 1200 2000 0 0 303 505}
```

Attributes

Name	*Type*	*Description*
LocalFiducialId	Id	A unique number for each instance of the LocalFiducial object.
ReferenceDesignator	String	The reference designator for this fiducial.
PositionX	Distance	The distance in X from the Location origin to this fiducial origin, with respect to the Location coordinate system.
PositionY	Distance	The distance in Y from the Location origin to this fiducial origin, with respect to the Location coordinate system.
PositionZ	Distance	The distance in Z from the Location origin to this fiducial origin, with respect to the Location coordinate system.
RotationZ	Angle	The rotation of this fiducial coordinate system with respect to the Z-axis of the Location coordinate system.
ReferenceLocationId	Id	The Id number of the corresponding Location for this fiducial.
ReferenceShapeId	Id	The Id number of the Shape used for this fiducial. The fiducial coordinate system defines the position and orientation of the shape coordinate system.

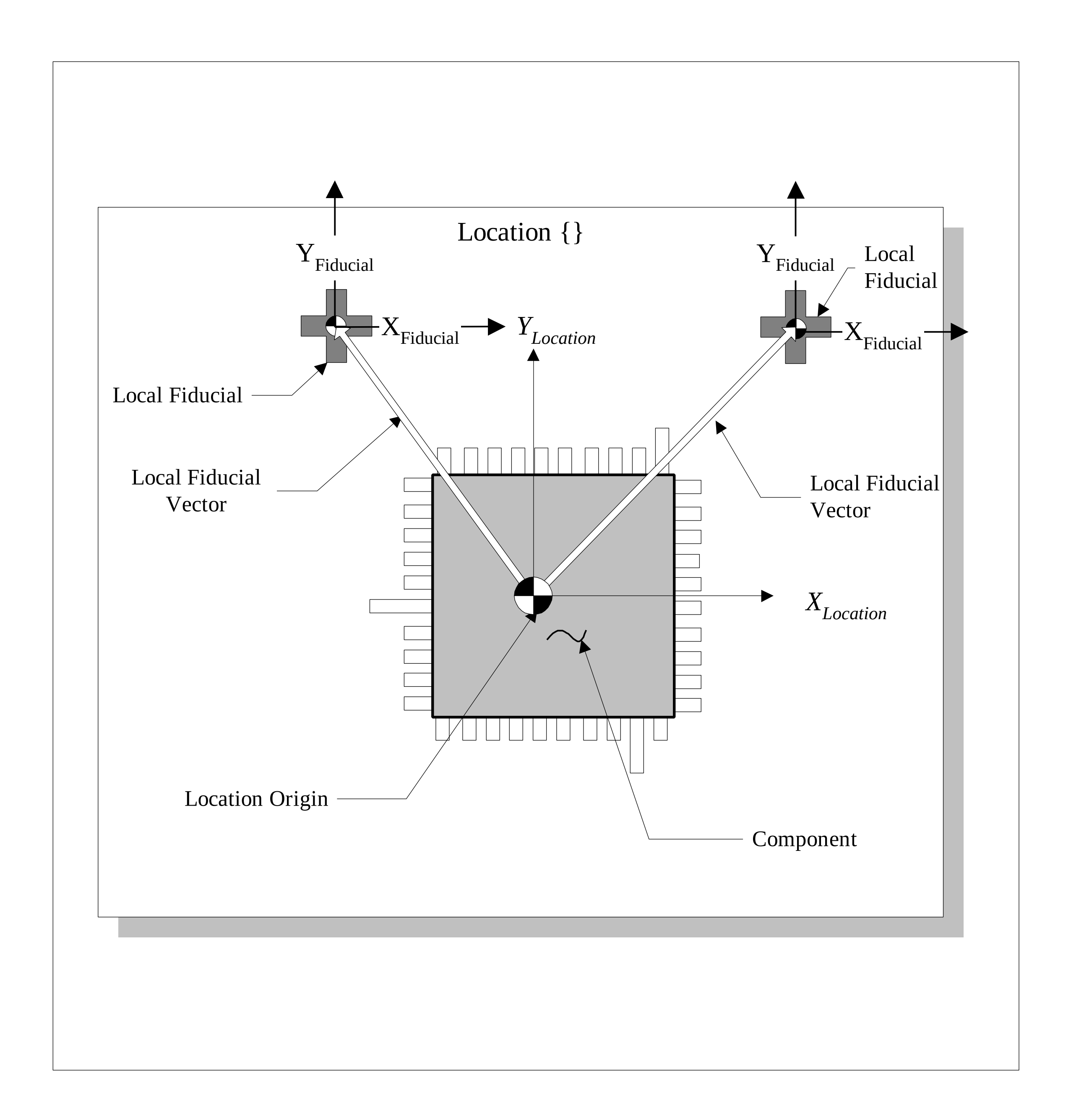


Figure D.4 A Location with corresponding LocalFiducials.

Location

Common Group
Product Category

Description Used to define the position and orientation of a component coordinate system with respect to an ImageDefinition coordinate system.

Schema Entry

{Location
{Id LocationId}
{String ReferenceDesignator}
{Distance PositionX}
{Distance PositionY}
{Distance PositionZ}
{Angle RotationZ}
{Id ReferenceComponentDefinitionId}
{Id ReferenceImageDefinitionId}
}

Data Example

{Location 230 “R1” 1200 1400 0 0 180 120}

Attributes

Name	***Type***	***Description***
LocationId	Id	A unique number for each instance of the Location object.
ReferenceDesignator	String	The reference designator for this Location.
PositionX	Distance	The distance in X from the Image origin to this Location origin, with respect to the ImageDefinition coordinate system.
PositionY	Distance	The distance in Y from the Image origin to this Location origin, with respect to the ImageDefinition coordinate system.
PositionZ	Distance	The distance in Z from the Image origin to this Location origin, with respect to the ImageDefinition coordinate system.
RotationZ	Angle	The rotation of this Location coordinate system with respect to the Z-axis of the ImageDefinition coordinate system.
ReferenceComponentDefinitionId	Id	The Id number of the corresponding ComponentDefinition for this Location.
ReferenceImageDefinitionId	Id	The Id number of the corresponding ImageDefinition for this Location.

Notes

1. A part name can also be linked to the component origin through a ComponentDefinition.
2. The definition of a component origin and its orientation are listed below.

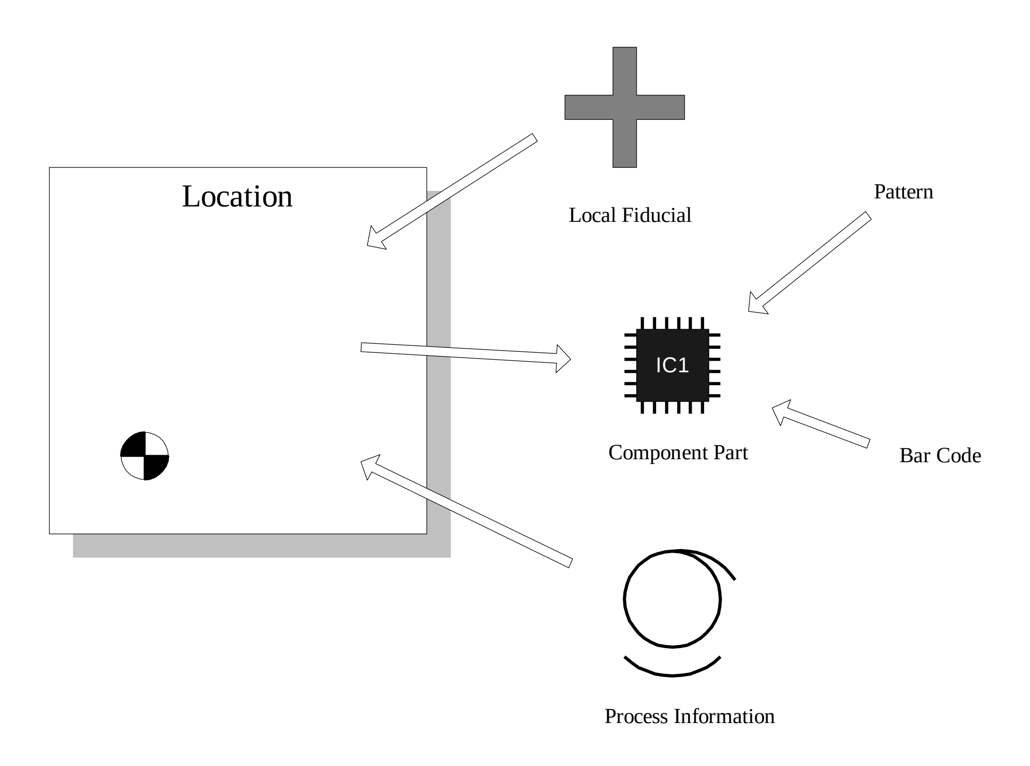


Figure D.5 A Location with corresponding linkages.

LocationGroup

Common Group
Process Category

Description A list of Locations and corresponding Images.

Schema Entry

```
{LocationGroup
{Id LocationGroupId}
{LIST {
{Id ReferenceImageId}
{Id ReferenceLocationId}
}}}
```

Data Example

```
{LocationGroup 220
{
230 170
230 123
}}
```

Attributes

Name	***Type***	***Description***
LocationGroupId	Id	A unique number for each instance of the LocationGroup object.
ReferenceImageId	Id	The Id number of the corresponding Image.
ReferenceLocationId	Id	The Id number of the corresponding Location.

LocationGroupOrdered

Common Group
Process Category

Description A list of Locations and corresponding Images where the order is significant.

Schema Entry

```
{LocationGroupOrdered
{Id LocationGroupOrderedId}
{LIST {
{Id ReferenceImageId}
{Id ReferenceLocationId}
}}}
```

Data Example

```
{LocationGroupOrdered 230
{
230 170
230 172
230 176
}}
```

Attributes

Name	***Type***	***Description***
LocationGroupOrderedId	Id	A unique number for each instance of the LocationGroupOrdered object.
ReferenceImageId	Id	The Id number of the corresponding Image.
ReferenceLocationId	Id	The Id number of the corresponding Location.

Panel

Common Group
Product Category

Description Used to define the dimensions of a Panel.

Schema Entry

{Panel
{Id PanelId}
{Distance LengthX}
{Distance LengthY}
{Distance LengthZ}
}

Data Example

{Panel 240 11900 149000 500}

Attributes

Name	***Type***	***Description***
PanelId	Id	A unique number for each instance of the Panel object.
LengthX	Distance	The length of the Panel in X. Values are limited to positive numbers.
LengthY	Distance	The length of the Panel in Y. Values are limited to positive numbers.
LengthZ	Distance	The length of the Panel in Z. Values are limited to positive numbers.

Notes

1. The position and orientation of the panel coordinate system is defined by the Image with ImageId = 0.
2. Only one panel object instance is allowed per file.

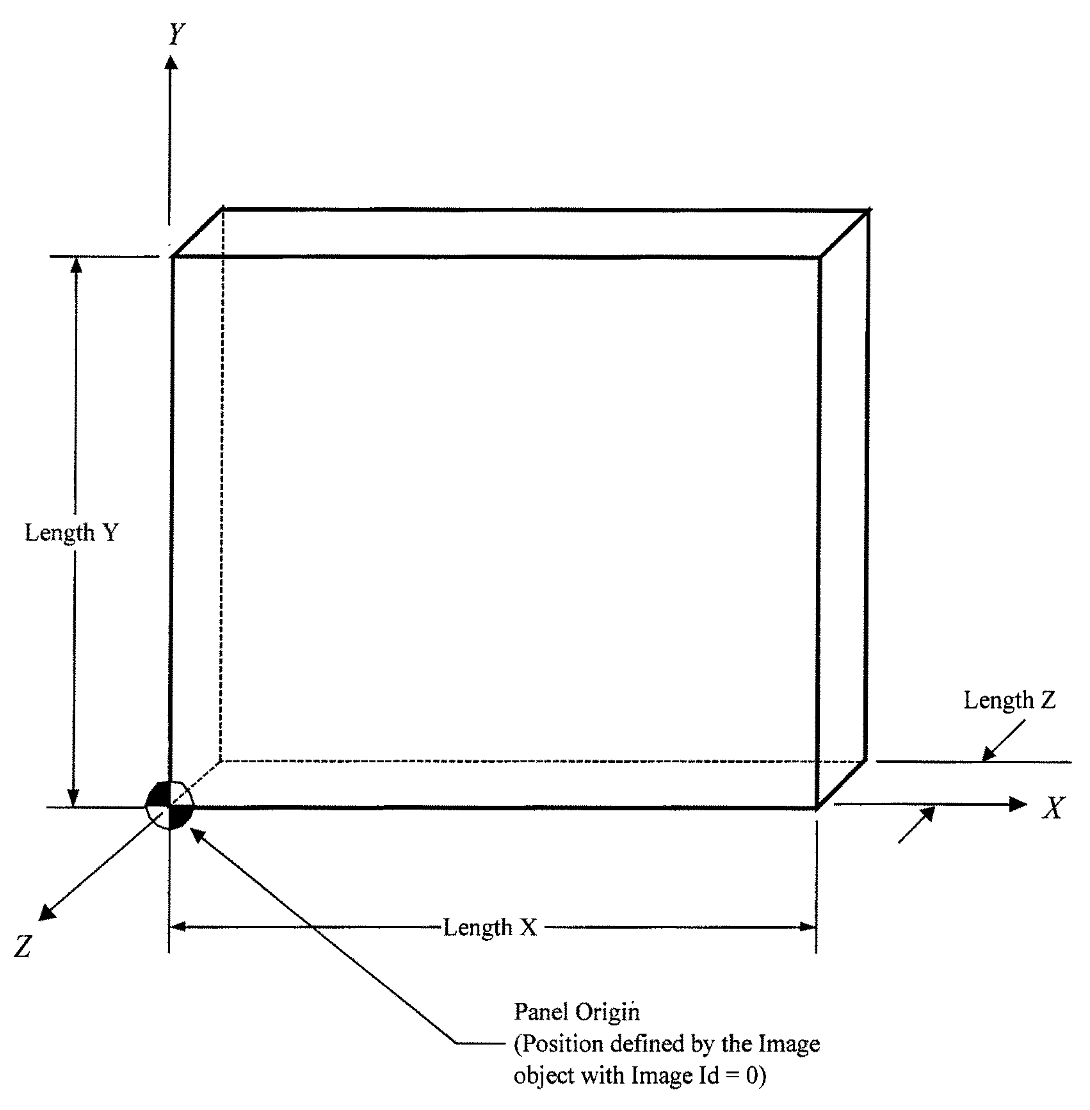


Figure D.6 The dimensions of the panel.

Pattern

Common Group
Product Category

Description Used to link a PatternDefinition to a part name.

Schema Entry

```
{Pattern
{Id PatternId}
{Id ReferenceComponentLinkId}
{Id ReferencePatternDefinitionId}
}
```

Data Example

```
{Pattern 250 130 260 }
```

Attributes

Name	***Type***	***Description***
PatternId	Id	A unique number for each instance of the Pattern object.
ReferenceComponentLinkId	Id	The Id number of the corresponding ComponentLink for this Pattern.
ReferencePatternDefinitionId	Id	The Id number of the corresponding PatternDefinition for this Pattern.

Notes

1. The position and orientation of a pattern origin are defined by a Location coordinate system.

PatternDefinition

Description Used to define a geometric relationship of features.

Schema Entry

```
{PatternDefinition
{Id PatternDefinitionId}
{String PatternDefinitionName}
{LIST {
{Object Feature}
}}}
```

Data Example

```
{PatternDefinition
{
{"Pad Pattern for 256 QFP"
{Feature 101 "Fist Pad" 230 300 0 180 103}
{Feature 102 "Second Pad" 230 320 0 180 103}
{Feature 103 "Third Pad" 230 340 0 180 103}
{Feature 104 "Fourth Pad" 230 360 0 180 103}
}
```

Attributes

Name	***Type***	***Description***
PatternDefinitionId	Id	A unique number for each instance of the PatternDefinition object.
PatternDefinitionName	String	The name of the PatternDefinition.
Feature	Object	An embedded Feature object.

Notes

1. The position and orientation of a Pattern coordinate system are defined by a Location coordinate system.

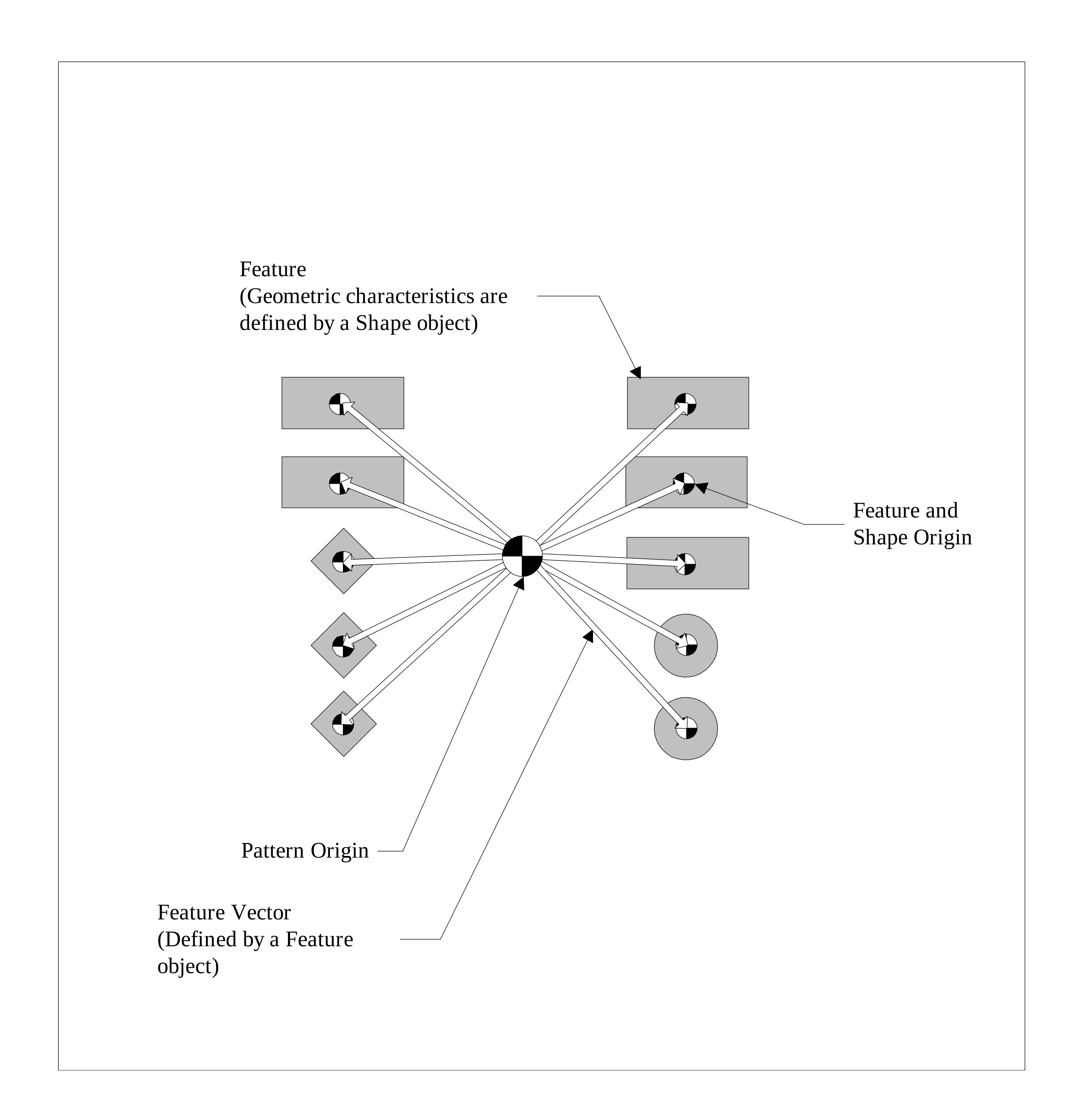


Figure D.7 A PatternDefinition with corresponding features.

Shape

Common Group
Product Category

Description Used to indicate a geometry type.

Schema Entry

```
{Shape
{Id ShapeId}
{SELECT {
{Object Cross}
{Object Diamond}
{Object Disc}
{Object Donut}
{Object Rectangle}
{Object Triangle}
{Object VendorShapeLink}
}}}
```

Data Example

```
{Shape 270
{Triangle 101 12 12 0}
}
```

Attributes

Name	***Type***	***Description***
ShapeId	Id	A unique number for each instance of the Shape object.
Cross	Object	An embedded Cross object.
Diamond	Object	An embedded Diamond object.
Disc	Object	An embedded Disc object.
Donut	Object	An embedded Donut object.
Rectangle	Object	An embedded Rectangle object.
Triangle	Object	An embedded Triangle object.
VendorShapeLink	Object	An embedded VendorShapeLink object.

SkipMark

Description Used to define an Image SkipMark. Defines the shape, position, and orientation of the SkipMark.

Schema Entry

{SkipMark
{Id SkipMarkId}
{Distance PositionX}
{Distance PositionY}
{Distance PositionZ}
{Angle RotationZ}
{Id ReferenceShapeId}
}

Data Example

{SkipMark 280 202.3 303.5 101.5 90.2 2}

Attributes

Name	*Type*	*Description*
SkipMarkId	Id	A unique number for each instance of the SkipMark object.
PositionX	Distance	The distance in X from the Image origin to this SkipMark origin, with respect to the Image coordinate system.
PositionY	Distance	The distance in Y from the Image origin to this SkipMark origin, with respect to the Image coordinate system.
PositionZ	Distance	The distance in Z from the Image origin to this SkipMark origin, with respect to the Image coordinate system.
RotationZ	Angle	The rotation of this SkipMark coordinate system with respect to the Z-axis of the Image coordinate system.
ReferenceShapeId	Id	The Id number of the Shape used for this SkipMark.

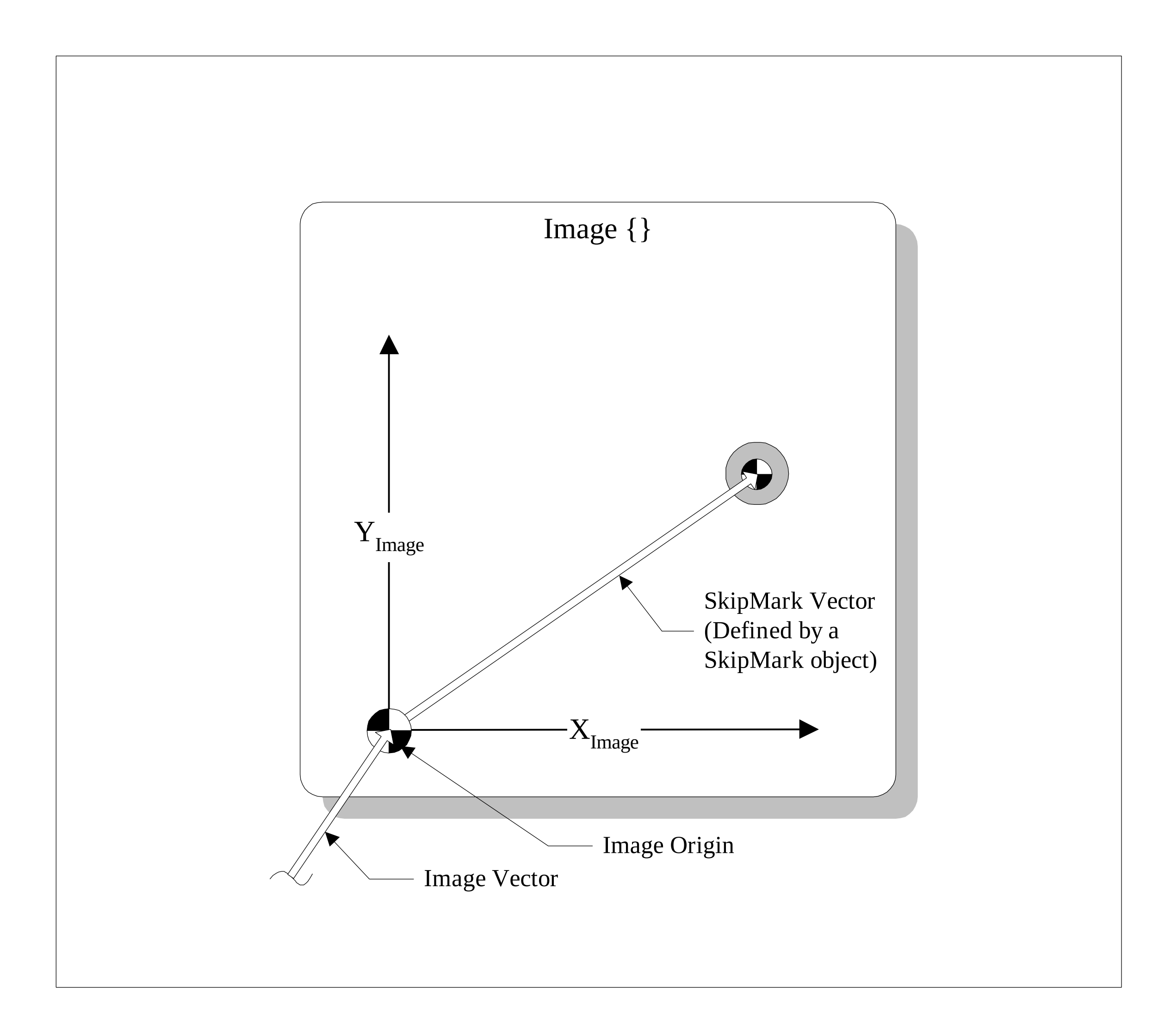


Figure D.8 An Image with corresponding SkipMark.

SRFFVersion

Common Group
Product Category

Description Indicates the SMEMA SRFF version used to create the file.

Schema Entry

```
{SRFFVersion
{Id SRFFVersionId}
{String VersionName}
}
```

Data Example

```
{SRFFVersion 1 "1.0"}
```

Attributes

Name	*Type*	*Description*
SRFFVersionId	Id	A unique number for each instance of the SRFFVersion object.
VersionName	String	The SRFF version used to create the file. The only value that can be used is: "1.0"

Notes

1. Only one instance of this object is allowed.

VendorShapeLink

Common Group
Product Category

Description Used to link a vendor-defined shape to the Shape object.

Schema Entry

{VendorShapeLink
{Id VendorShapeLinkId}
}

Data Example

Attributes

Name	*Type*	*Description*
VendorShapeLinkId	Id	A unique number for each instance of the VendorShapeLink object.

DispenseOrder

Description A list of grouped Features where the order of the groups is significant with respect to dispense operations.

Schema Entry

```
{DispenseOrder
{Id DispenseOrderId}
{LIST {
{SELECT {
{Object FeatureGroup}
{Object FeatureGroupOrdered}
}}}}}
```

Data Example

```
{DispenseOrder 1
{
{FeatureGroup 140 {2 3 12 22 2 3 12 23 2 3 12 24 2 3 12 25}
{FeatureGroupOrdered 110 {3 2 3 12 22 2 3 12 23 2 3 12 24 2 3 12 25}
{FeatureGroup 141 {2 3 12 25 2 3 12 26 2 3 12 27 2 3 12 28}
}}
```

Attributes

Name	*Type*	*Description*
DispenseOrderId	Id	A unique number for each instance of the DispenseOrder object.
FeatureGroup	Object	An embedded FeatureGroup object.
FeatureGroupOrdered	Object	An embedded FeatureGroupOrdered object.

Notes

1. At most one instance of the DispenseOrder object is allowed in the vendor independent data section.
2. An instance of DispenseOrder in a vendor specific data section supercedes an instance in the vendor independent data section.
3. The order of Features within a FeatureGroup is not significant.
4. The order of Features within a FeatureGroupOrdered is significant.

InspectOrder

Description A list of grouped Features where the order of the groups is significant with respect to inspection operations.

Schema Entry

```
{InspectOrder
{Id InspectOrderId}
{LIST {
{SELECT {
{Object FeatureGroup}
{Object FeatureGroupOrdered}
}}}}}
```

Data Example

```
{InspectOrder 1
{
{FeatureGroup 140 {2 3 12 22 2 3 12 23 2 3 12 24 2 3 12 25}
{FeatureGroupOrdered 110 {3 2 3 12 22 2 3 12 23 2 3 12 24 2 3 12 25}
{FeatureGroup 141 {2 3 12 25 2 3 12 26 2 3 12 27 2 3 12 28}
}}
```

Attributes

Name	***Type***	***Description***
InspectOrderId	Id	A unique number for each instance of the InspectOrder object.
FeatureGroup	Object	An embedded FeatureGroup object.
FeatureGropuOrdered	Object	An embedded FeatureGroupOrdered object.

Notes

1. At most one instance of the InspectOrder object is allowed in the vendor independent data section.
2. An instance of InspectOrder in a vendor specific data section supercedes an instance in the vendor independent data section.
3. The order of Features within a FeatureGroup is not significant.
4. The order of Features within a FeatureGroupOrdered is significant.

PlacementOrder

Placement Group
Process Category

Description

A list of grouped Locations where the order of the groups is significant with respect to placement operations.

Schema Entry

```
{PlacementOrder
{Id PlacementOrderId}
{LIST {
{SELECT {
{Object LocationGroup}
{Object LocationGroupOrdered}
}}}}}
```

Data Example

```
{PlacementOrder 1
{
{LocationGroup 221 {330 170 330 123}}
{LocationGroup 222 {130 170 130 123}}
{LocationGroupOrdered 531 {340 171  230 171  230 176}}
{LocationGroupOrdered 532 {342 172  230 172  230 177}}
{LocationGroupOrdered 530 {343 173  230 173  230 178}}
{LocationGroup 123 {980 171 232 123}}
{LocationGroup 101 {982 172 232 124}}
{LocationGroupOrdered 130 {260 170  230 172  230 176}}
{LocationGroup 501 {1001 1 23 12}}
}}
```

Attributes

Name	***Type***	***Description***
PlacementOrderId	Id	A unique number for each instance of the PlacementOrder object.
LocationGroup	Object	An embedded LocationGroup object
LocationGroupOrdered	Object	An embedded LocationGroupOrdered object.

Notes

1. At most one instance of the PlacmentOrder object is allowed in the vendor independent data section.
2. An instance of PlacementOrder in a vendor specific data section supercedes an instance in the vendor independent data section.
3. The order of Locations within a LocatoinGroup is not significant.
4. The order of Locations within a LocationGroupOrdered is significant.

Print

Print Group
Process Category

Description A process object defining the print stroke action parameters.

Schema Entry

```
{Print
{Id PrintId}
{String PrintOperation}
{Int PrintDeposits}
{Distance SnapOffZ}
{Velocity SeparationSpeed}
{Distance SeparationDistance}
{Distance MarginAreaLengthX}
{Distance MarginAreaLengthY}
{Distance MarginAreaOffsetX}
{Distance MarginAreaOffsetY}
{Id ReferenceImageId}
}
```

Data Example {Print 1 “PrintPrint” 1 0 30 2.5 30 40 10 12 2}

Attributes

Name	***Type***	***Description***
PrintId	Id	A unique number for each instance of the Print object.
PrintOperation	String	The type of print operation to be performed. Values are limited to the following strings: "PrintPrint" "PrintFlood" "FloodPrint"
PrintDeposits	Int	The number of Print strokes to be performed on a single panel.
SnapOffZ	Distance	The distance in Z between the panel and the screen during the print stroke.
SeparationSpeed	Velocity	The velocity of separation between the panel and the screen after the print has completed.
SeparationDistance	Distance	The distance in Z between the panel and the screen during the separation process. The separation speed is applied for this distance.
MarginAreaLengthX	Distance	The distance in X of the print margin area with respect to the ReferenceImage coordinate system.
MarginAreaLengthY	Distance	The distance in Y of the print margin area with respect to the ReferenceImage coordinate system.
MarginAreaOffsetX	Distance	The offset in X of the print margin area origin with respect to the reference image origin.
MarginAreaOffsetY	Distance	The offset in Y of the print margin area origin with respect to the reference image origin.
ReferenceImageId	Id	The Id number of the reference image.

PrinterAlignment

Print Group
Process Category

Description A process object defining alignment fiducial information for screen printers.

Schema Entry

```
{PrinterAlignment
{Id PrinterAlignmentId}
{LIST  {
{Id ReferenceImageFiducialId}
{Id ReferenceScreenFiducialId}
}}}
```

Data Example

```
{PrinterAlignment 1 {1 2 1 3 1 5 1 4}}
```

Attributes

Name	***Type***	***Description***
PrinterAlignmentId	Id	A unique number for each instance of the PrinterAlignment object.
ReferenceImageFiducialId	Id	The Id of the corresponding ImageFiducial.
ReferenceScreenFiducialId	Id	The Id of the corresponding ScreenFiducial.

PrintArea

Print Group
Product Category

Description Product information defining the area to be printed.

Schema Entry

```
{PrintArea
{Id PrintAreaId}
{Distance PrintAreaLengthX}
{Distance PrintAreaLengthY}
{Id ReferenceImageId}
}
```

Data Example

```
{PrintArea 1 100 100 12}
```

Attributes

Name	***Type***	***Description***
PrintAreaId	Id	A unique number for each instance of the PrintArea object.
PrintAreaLengthX	Distance	The length of the area that must be printed along the x-axis of the reference Image.
PrintAreaLengthY	Distance	The length of the area that must be printed along the y-axis of the reference Image.
ReferenceImageId	Id	The Id number of the reference image.

ScreenProperties

Print Group
Product Category

Description A product object that defines the dimensions of a screen stencil, mesh, and frame.

Schema Entry

{ScreenProperties
{Id ScreenPropertiesId}
{String ScreenRecordName}
{Distance FrameOuterLengthX}
{Distance FrameOuterLengthY}
{Distance StencilOriginOffsetX}
{Distance StencilOriginOffetY}
{Distance StencilOuterLengthX}
{Distance StencilOuterLengthY}
{Distance MeshOriginOffsetX}
{Distance MeshOriginOffetY}
{Distance MeshOuterLengthX}
{Distance MeshOuterLengthY}
{Distance ScreenImageOriginOffsetX}
{Distance ScreenImageOriginOffetY}
{Distance ScreenImageOuterLengthX}
{Distance ScreenImageOuterLengthY}
{Id ReferenceImageId}
{Id ReferenceBarCodeId}
}

Data Example

{ScreenProperties 1 "265 Front justified" 500 550 10 20 300 350 20 30 200 250 65 70 150 175 11 10}

Attributes

Name	*Type*	*Description*
ScreenPropertiesId	Id	A unique number for each instance of the ScreenProperties object.
ScreenRecordName	String	A unique name for this screen.
FrameOuterLengthX	Distance	The outer length of the frame in the X dimension with respect to the frame coordinate system.
FrameOuterLengthY	Distance	The outer length of the frame in the Y dimension with respect to the frame coordinate system.
MeshOriginOffsetX	Distance	The offset in X of the mesh origin with respect to the frame origin.
MeshOriginOffsetY	Distance	The offset in Y of the mesh origin with respect to the frame origin.
MeshOuterLengthX	Distance	The outer length of the mesh in the X dimension with respect to the frame coordinate system.
MeshOuterLengthY	Distance	The outer length of the mesh in the Y dimension with respect to the frame coordinate system.
StencilOriginOffsetX	Distance	The offset in X of the stencil origin with respect to the frame origin.

StencilOriginOffsetY	Distance	The offset in Y of the stencil origin with respect to the frame origin
StencilOuterLengthX	Distance	The outer length of the stencil in the X dimension with respect to the frame coordinate system.
StencilOuterLengthY	Distance	The outer length of the stencil in the Y dimension with respect to the frame coordinate system.
ScreenImageOuterLengthX	Distance	The outer length of the screen image in the X dimension with respect to the frame coordinate system.
ScreenImageOuterLengthY	Distance	The outer length of the screen image in the Y dimension with respect to the frame coordinate system.
ScreenImageOriginOffsetX	Distance	The offset in X of the screen image origin with respect to the frame origin.
ScreenImageOriginOffsetY	Distance	The offset in Y of the screen image origin with respect to the frame origin.
ReferenceImageId}	Id	The Id number of the reference image. The reference image defines the coordinate system of the frame.
ReferenceBarCodeId	Id	The Id number of the corresponding bar code.

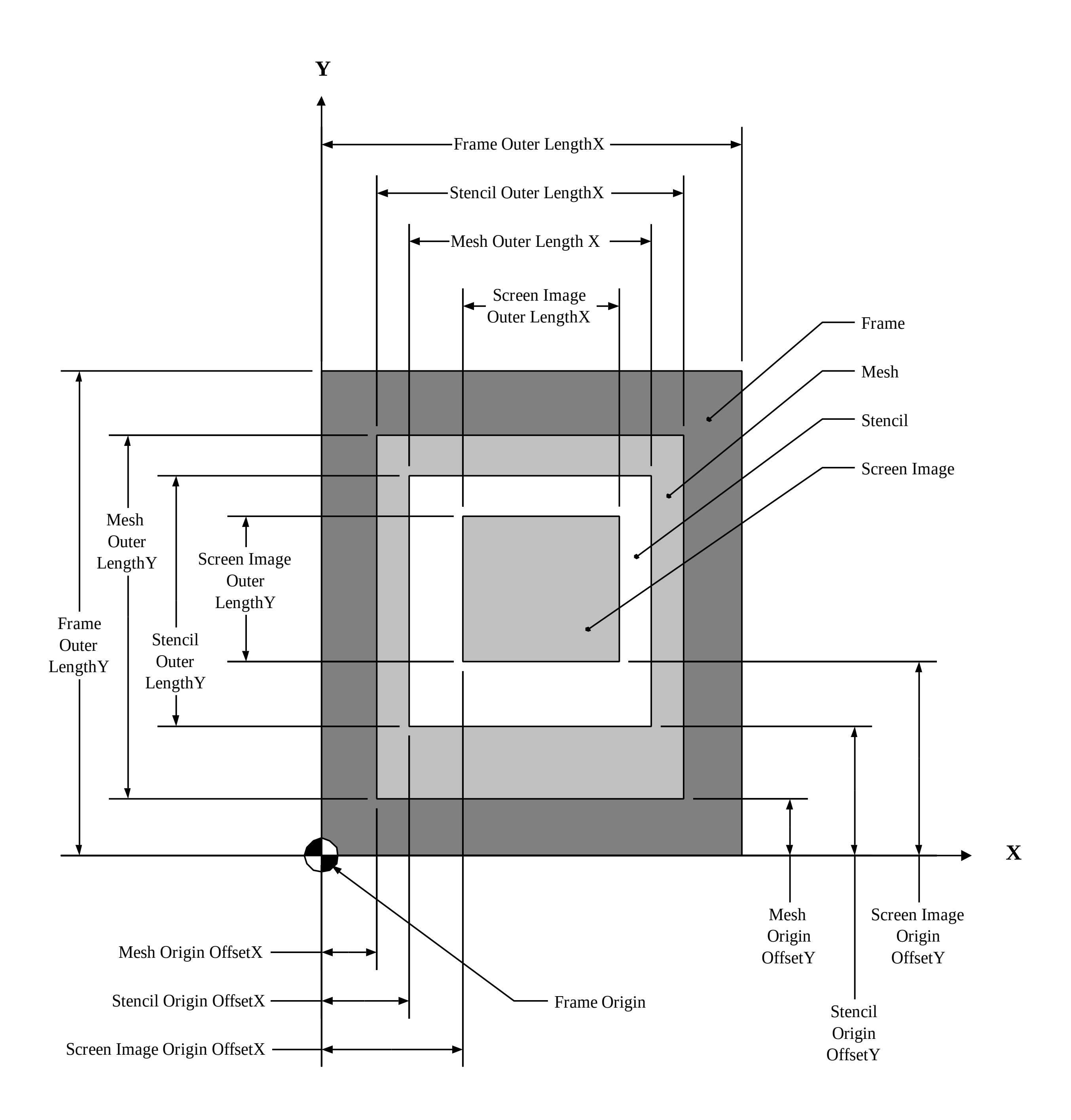


Figure D.9 A print screen with labeled attributes.

ScreenFiducial

Print Group
Product Category

Description A product object defining a fiducial on a printer screen.

Schema Entry

```
{ScreenFiducial
{Id ScreenFiducialId}
{String ReferenceDesignator}
{Distance PositionX}
{Distance PositionY}
{Distance PositionZ}
{Angle RotationZ}
{Id ReferenceImageId}
{Id ReferenceShapeId}
}
```

Data Example

{ScreenFiducial 1 “Screen Fid 1” 1 1 100 100 0 0 1}

Attributes

Name	***Type***	***Description***
ScreenFiducialId	Id	A unique number for each instance of the ScreenFiducial object.
ReferenceDesignator	String	The name of the fiducial.
PositionX	Distance	The position in X from the origin of the Image object that the fiducial is referenced to.
PositionY	Distance	The position in Y from the origin of the Image object that the fiducial is referenced to.
PositionZ	Distance	The position in Z from the origin of the Image object that the fiducial is referenced to.
RotationZ	Angle	The rotation of the shape object when used for this screen fiducial.
ReferenceImageId	Id	The Id number of the reference Image.
ReferenceShapeId	Id	The reference Id of the shape object for the fiducial.

Squeegee

Print Group
Process Category

Description Process settings for a squeegee.

Schema Entry

```
{Squeegee
{Id SqueegeeId}
{Force SqueegeePressure}
{Velocity PrintSpeed}
{Velocity FloodSpeed}
{Velocity SqueegeeLiftSpeed}
{Distance SqueegeeLiftDistance}
{Angle SqueegeeRake}
{Distance FloodHeight}
{Time SqueegeeLiftDelay}
}
```

Data Example

```
{Squeegee 2 10 50 * 30 50 60 * 0}
```

Attributes

Name	*Type*	*Description*
SqueegeeId	Id	A unique number for each instance of the Squeegee object.
SqueegeePressure	Force	The force that the squeegee will apply during printing.
PrintSpeed	Velocity	The velocity of the squeegee across the image during the print stroke.
FloodSpeed	Velocity	The velocity of the flood bar across the image during the flood stroke.
SqueegeeLiftSpeed	Velocity	The velocity that the squeegee will use to lift away from the screen at the end of the print stroke.
SqueegeeLiftDistance	Distance	The distance in Z that the squeegee will move away from the screen at the end of the print stroke.
SqueegeeRake	Angle	The angle between the squeegee blade and the screen during the print stroke.
FloodHeight	Distance	The distance in Z between the flood blade and the screen during a flood stroke.
SqueegeeLiftDelay	Time	The delay before the squeegee lifts from the screen at the end of a print stroke.

SqueegeeProperties

Print Group
Process Category

Description Used to define the properties of a squeegee.

Schema Entry

```
{SqueegeeDefinition
{Id SqueegeeDefinitionId}
{String SqueegeeRecordName}
{Distance SqueegeeBladeHeight}
{Angle SqueegeeRecordAngle}
{Distance SqueegeeBladeOffset}
{Distance SqueegeeLength}
}
```

Data Example

{SqueegeeDefinition 1 “100 mm 45 degree metal” 5 30 2 7}

Attributes

Name	***Type***	***Description***
SqueegeeDefinitionId	Id	A unique number for each instance of the SqueegeeDefinition object.
SqueegeeRecordName	String	A unique name used to identify a squeegee.
SqueegeeBladeHeight	Distance	The distance in Z from the mounting face of the squeegee to the tip of the squeegee blade.
SqueegeeRecordAngle	Angle	The angle that the squeegee blade is fixed to the squeegee holder. This is measured as the angle between the center line of the squeegee holder as viewed from the side and the working face of the squeegee blade.
SqueegeeBladeOffset	Distance	The distance between the center line of the squeegee holder when viewed from the side and the tip of the squeegee blade.
SqueegeeLength	Distance	The length of the squeegee measured at the tip of the blade.

Cross

Description Used to define the two dimensional geometry of a cross shape.

Schema Entry

```
{Cross
{Id CrossId}
{Distance Base}
{Distance Height}
{Distance BaseLegWidth}
{Distance HeightLegWidth}
}
```

Data Example

```
{Cross 101 10 10 2 2}
```

Attributes

Name	*Type*	*Description*
CrossId	Id	A unique number for each instance of the Cross object.
Base	Distance	The base of the Cross.
Height	Distance	The height of the Cross.
BaseLegWidth	Distance	The width of the base leg of the Cross
HeightLegWidth	Distance	The width of the height leg of the Cross.

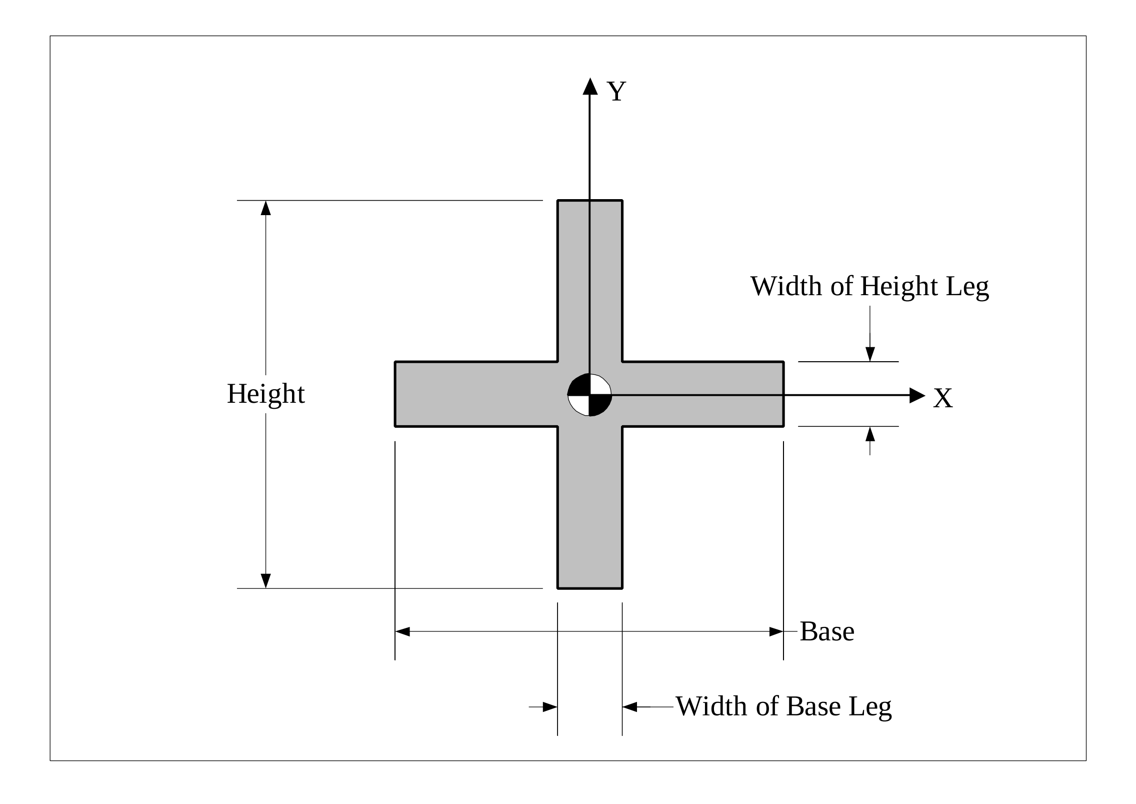


Figure D.10 Dimensions used to define a cross.

Diamond

Description	Used to define the two dimensional geometry of a diamond shape.
Schema Entry	{Diamond {Id DiamondId} {Distance Base} {Distance Height} }
Data Example	{Diamond 1 25 25}

Attributes

Name	***Type***	***Description***
DiamondId	Id	A unique number for each instance of the Diamond object.
Base	Distance	The base of the Diamond.
Height	Distance	The height of the Diamond.

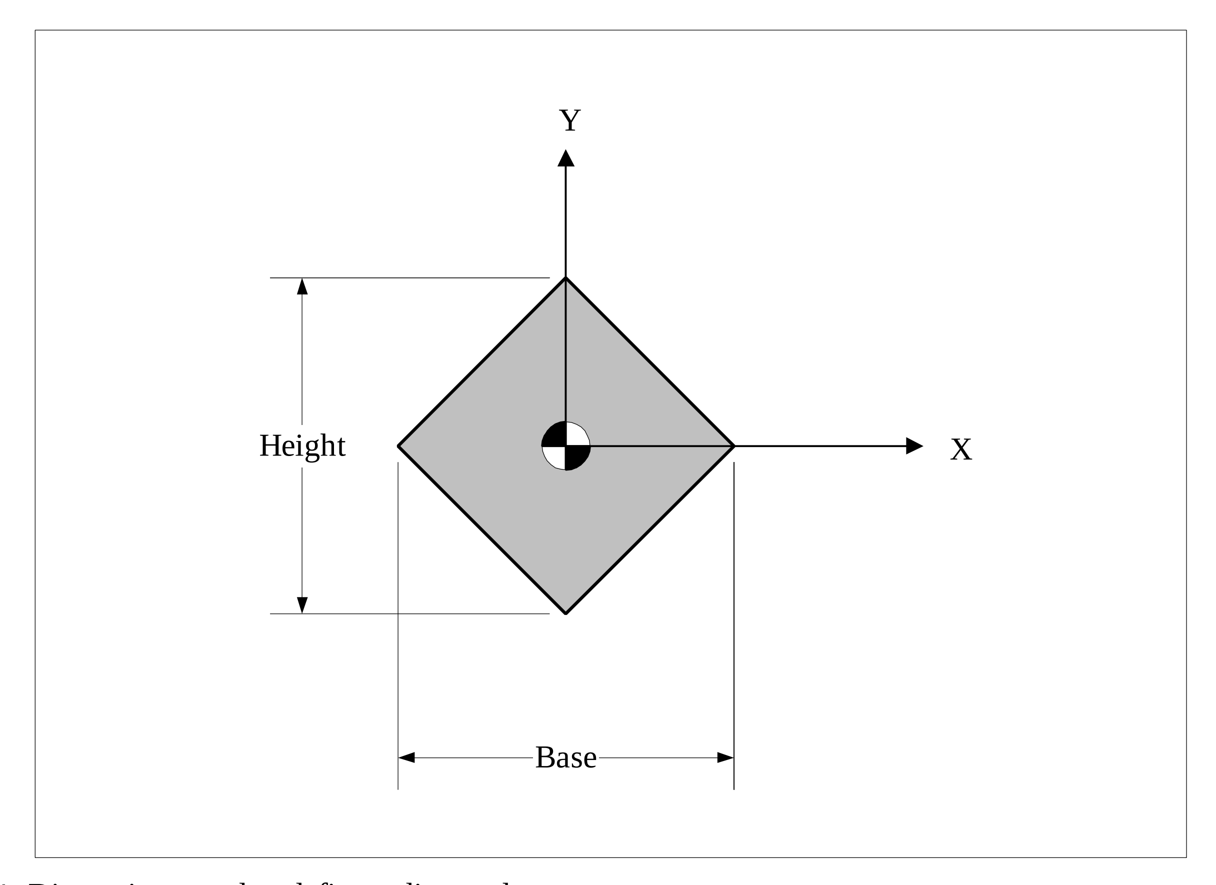


Figure D.11 Dimensions used to define a diamond.

Disc

Description Used to define the two dimensional geometry of a disc shape.

Schema Entry

```
{Disc
{Id DiscId}
{Distance Diameter}
}
```

Data Example

```
{Disc 2 20}
```

Attributes

Name	*Type*	*Description*
DiscId	Id	A unique number for each instance of the Disc object.
Diameter	Distance	The diameter of the Disc.

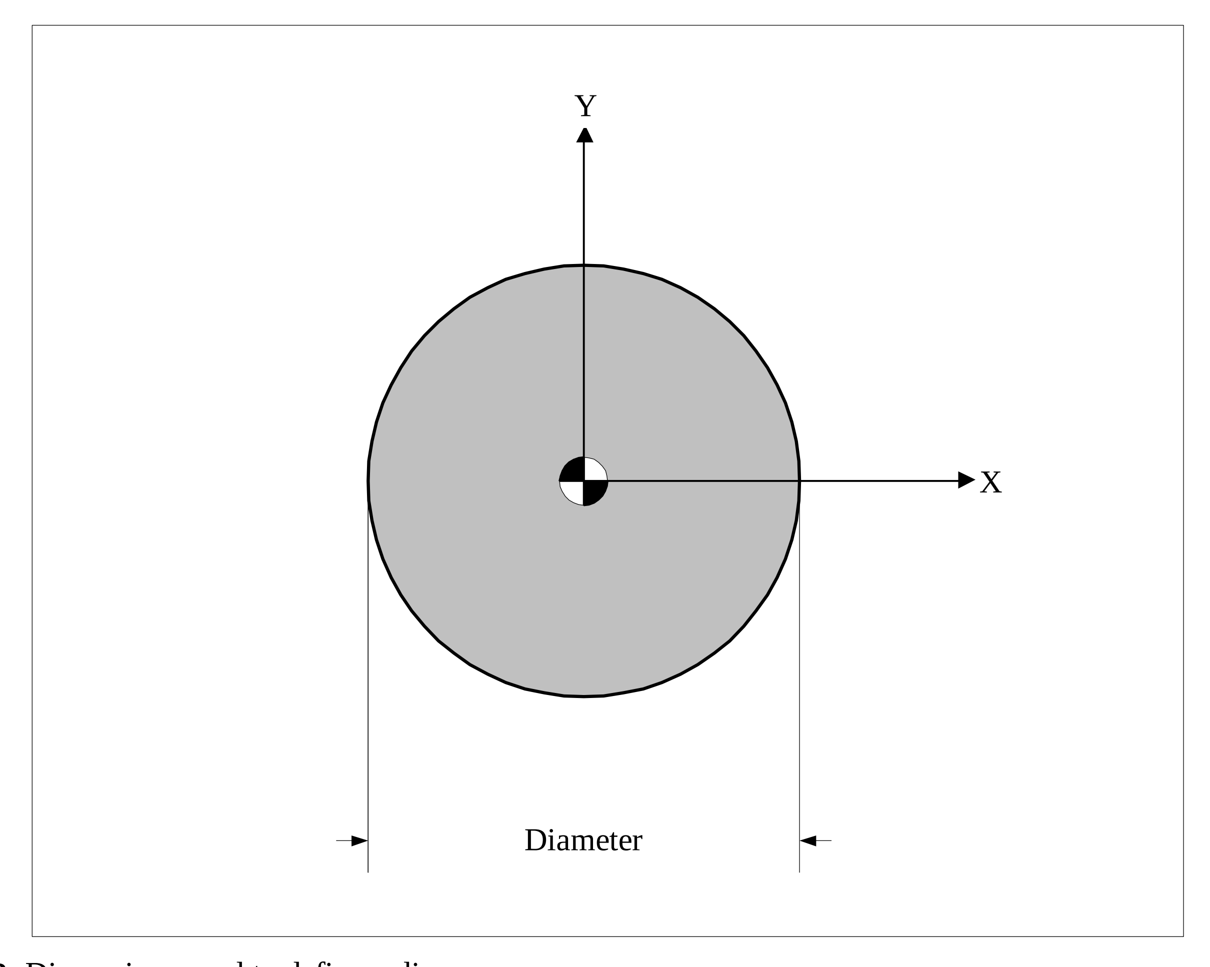


Figure D.12 Dimensions used to define a disc.

Donut

Description Used to define the two dimensional geometry of a donut shape.

Schema Entry

{Donut
{Id DonutId}
{Distance InnerDiameter}
{Distance OuterDiameter}
}

Data Example

{Donut 2 15 25}

Attributes

Name	*Type*	*Description*
DonutId	Id	A unique number for each instance of the Donut object.
InnerDiameter	Distance	The inner diameter of the Donut.
OuterDiameter	Distance	The outer diameter of the Donut.

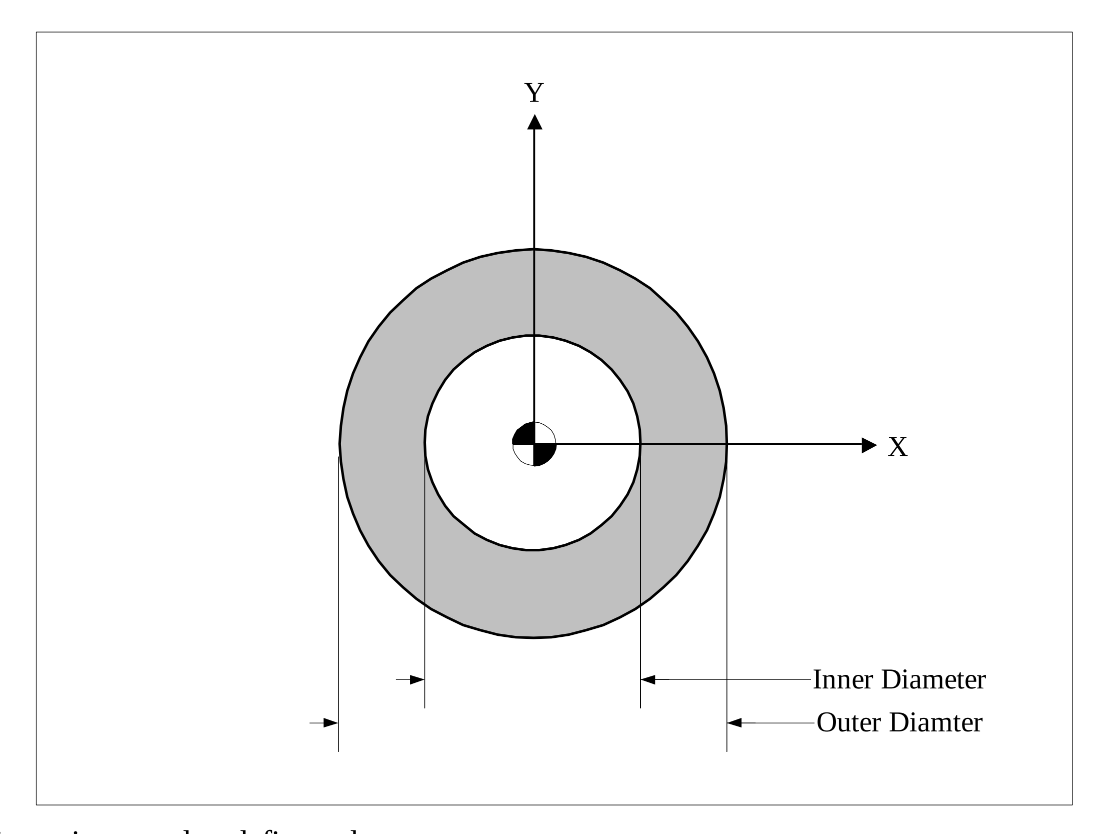


Figure D.13 Dimensions used to define a donut.

Rectangle

Description Used to define the two dimensional geometry of a rectangle shape.

Schema Entry

```
{Rectangle
{Id RectangleId}
{Distance Base}
{Distance Height}
}
```

Data Example

```
{Rectangle 101 245 345}
```

Attributes

Name	*Type*	*Description*
RectangleId	Id	A unique number for each instance of the Rectangle object.
Base	Distance	The base of the Rectangle.
Height	Distance	The height of the Rectangle.

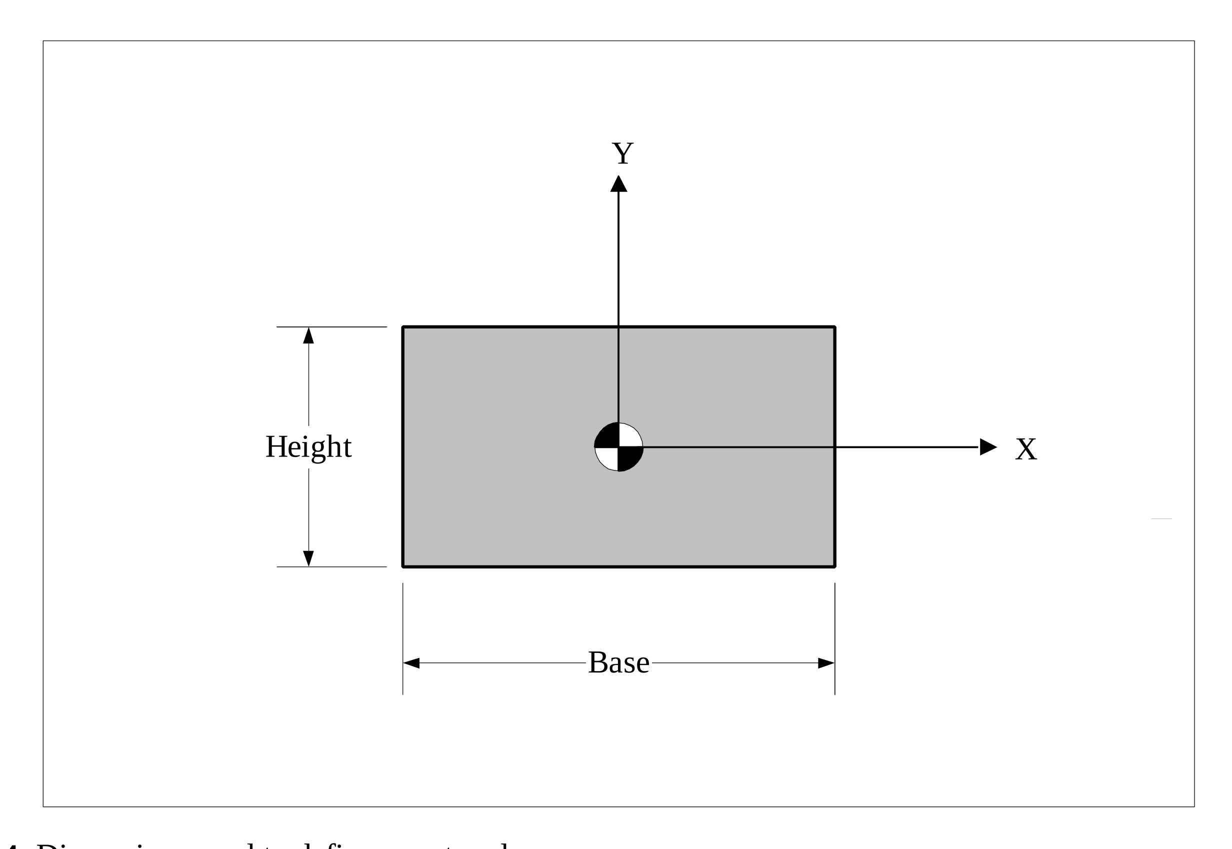


Figure D.14 Dimensions used to define a rectangle.

Triangle

Shape Group
Product Category

Description Used to define the two dimensional geometry of a triangle shape.

Schema Entry

{Triangle
{Id TriangleId}
{Distance Base}
{Distance Height}
{Distance Offset}
}

Data Example

{Triangle 101 10.5 10.5 3}

Attributes

Name	***Type***	***Description***
TriangleId	Id	A unique number for each instance of the Triangle object.
Base	Distance	The base of the Triangle.
Height	Distance	The height of the Triangle.
Offset	Distance	The distance along the X-axis from the left end of the base of the Triangle to the tip.

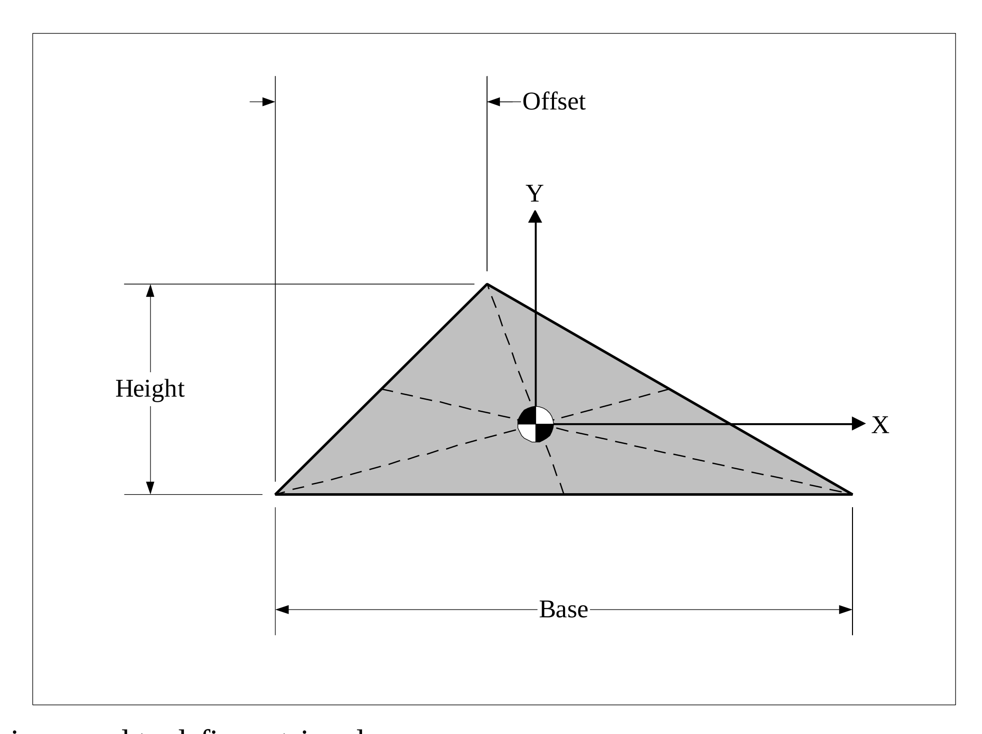


Figure D.15 Dimensions used to define a triangle.

AccelerationUnits

Units Group
Product Category

Description Used to define the units of acceleration.

Schema Entry

```
{AccelerationUnits
{Id AccelerationUnitsId}
{String UnitsOfAcceleration}
}
```

Data Example

```
{AccelerationUnits 1 "Meters/Second^2"}
```

Attributes

Name	*Type*	*Description*
AccelerationUnitsId	Id	A unique number for each instance of the AccelerationUnits object.
UnitsOfAcceleration	String	The default units of acceleration. Values are limited to the following strings: "Feet/Second^2" "(Inches/1000)/Minute^2" "(Inches/1000)/Second^2" "Inches/Minute^2" "Inches/Second^2" "Meters/Second^2"

Notes

1. Only one instance of the AcclerationUnits object is allowed in each file.

AngleUnits

Units Group
Product Category

Description Used to define the units of angular measurement.

Schema Entry

```
{AngleUnits
{Id AngleUnitsId}
{String UnitsOfAngularMeasurement}
}
```

Data Example

```
{AngleUnits 99 "Degrees"}
```

Attributes

Name	*Type*	*Description*
AngleUnitsId	Id	A unique number for each instance of the AngleUnits object.
UnitsOfAngularMeasurement	String	The default units of angle. Values are limited to the following strings: "Degrees/100" "Degrees/1000" "Degrees" "Minutes" "Quadrants" "Radians" "Seconds"

Notes

1. Only one instance of the AngleUnits object is allowed in each file.

AngularAccelerationUnits

Description Used to define the units of angular acceleration.

Schema Entry

```
{AngularAccelerationUnits
{Id AngularAccelerationId}
{String UnitsOfAngularAcceleration}
}
```

Data Example

```
{AngularAccelerationUnits 11 "Degrees/Second^2"}
```

Attributes

Name	*Type*	*Description*
AngularAccelerationId	Id	A unique number for each instance of the AngularAcceleration object.
UnitsOfAngularAcceleration	String	The default units of angular acceleration. Values are limited to the following strings: "Degrees/Second^2" "Radians/Second^2"

Notes

1. Only one instance of the AngularAcclerationUnits object is allowed in each file.

AngularVelocityUnits

Units Group
Product Category

Description Used to define the units of angular velocity.

Schema Entry

```
{AngularVelocityUnits
{Id AngularVelocityId}
{String UnitsOfAngularVelocity}
}
```

Data Example

```
{AngularVelocityUnits 1 "Radians/Second"}
```

Attributes

Name	*Type*	*Description*
AngularVelocityId	Id	A unique number for each instance of the AngularVelocity object.
UnitsOfAngularVelocity	String	The default units of angular velocity. Values are limited to the following strings: “Degrees/Second” “Radians/Second”

Notes

1. Only one instance of the AngularVelocityUnits object is allowed in each file.

DistanceUnits

Units Group
Product Category

Description Used to define the units of distance.

Schema Entry

```
{DistanceUnits
{Id DistanceUnitsId}
{String UnitsOfDistance}
}
```

Data Example

```
{DistanceUnits 101 "Millimeters"}
```

Attributes

Name	*Type*	*Description*
DistanceUnitsId	Id	A unique number for DistanceUnits object.
UnitsOfDistance	String	The default units of distance. Values are limited to the following strings: "Inches/1000" "Inches" "Meters" "Microns" "Millimeters/100" "Millimeters"

Notes

1. Only one instance of the DistanceUnits object is allowed in each file.

FlowUnits

Units Group
Product Category

Description Used to define the units of volumetric flow.

Schema Entry

```
{FlowUnits
{Id FlowUnitsId}
{String UnitsOfFlow}
}
```

Data Example

{FlowUnits 1 "Gallons/Minute"}

Attributes

Name	*Type*	*Description*
FlowUnitsId	Id	A unique number for FlowUnits object.
UnitsOfFlow	String	The default units of flow. Values are limited to the following strings: "CubicFeet/Hour" "CubicFeet/Minute" "CubicMeters/Hour" "CubicMeters/Minute" "Gallons/Minute" "Liters/Minute" "Milliliters/Minute" "Ounces/Minute"

Notes

1. Only one instance of the FlowUnits object is allowed in each file.

ForceUnits

Units Group
Product Category

Description Used to define the units of force.

Schema Entry

```
{ForceUnits
{Id ForceUnitsId}
{String UnitsOfForce}
}
```

Data Example

```
{ForceUnits 101 “Pounds”}
```

Attributes

Name	***Type***	***Description***
ForceUnitsId	Id	A unique number for ForceUnits object.
UnitsOfForce	String	The default units of force. Values are limited to the following strings: “Newtons” “Ounces” “Pounds”

Notes

1. Only one instance of the ForceUnits object is allowed in each file.

HumidityUnits

Units Group
Product Category

Description Used to define the units of humidity.

Schema Entry

```
{HumidityUnits
{Id HumidityUnitsId}
{String UnitsOfHumidity}
}
```

Data Example

```
{HumidityUnits 1 “DewPoint”}
```

Attributes

Name	***Type***	***Description***
HumidityUnitsId	Id	A unique number for each HumidityUnits object.
UnitsOfHumidity	String	The default units of humidity. Values are limited to the following strings: “DewPoint” “RelativeHumidity”

Notes

1. Only one instance of the HumidityUnits object is allowed in each file.

MassUnits

Units Group
Product Category

Description Used to define the units of mass.

Schema Entry

```
{MassUnits
{Id MassUnitsId}
{String UnitsOfMass}
}
```

Data Example

```
{MassUnits 3 "Milligrams"}
```

Attributes

Name	***Type***	***Description***
MassUnitsId	Id	A unique number for each MassUnits object.
UnitsOfMass	String	The default units of mass. Values are limited to the following strings: "Grams" "Kilograms" "Milligrams" "PoundsMass"

Notes

1. Only one instance of the MassUnits object is allowed in each file.

PowerUnits

Units Group
Product Category

Description Used to define the units of power.

Schema Entry

{PowerUnits
{Id PowerUnitsId}
{String UnitsOfPower}
}

Data Example

{PowerUnits 204 "BritishThermalUnits"}

Attributes

Name	*Type*	*Description*
PowerUnitsId	Id	A unique number for each PowerUnits object.
UnitsOfPower	String	The default units of power. Values are limited to the following strings: "BritishThermalUnits" "Watts"

Notes

1. Only one instance of the PowerUnits object is allowed in each file.

PressureUnits

Units Group
Product Category

Description Used to define the units of pressure.

Schema Entry

{PressureUnits
{Id PressureUnitsId}
{String UnitsOfPressure}
}

Data Example

{PressureUnits 101 "Pascals"}

Attributes

Name	*Type*	*Description*
PressureUnitsId	Id	A unique number for each PressureUnits object.
UnitsOfPressure	String	The default units of pressure. Values are limited to the following strings: "Bars" "Pascals" "Pounds/Inch^2"

Notes

1. Only one instance of the PressureUnits object is allowed in each file.

TemperatureUnits

Description Used to define the units of temperature.

Schema Entry

{TemperatureUnits
{Id TemperatureUnitsId}
{String UnitsOfTemperature}
}

Data Example

{TemperatureUnits 75 "DegreesFahrenheit"}

Attributes

Name	***Type***	***Description***
TemperatureUnitsId	Id	A unique number for each TemperatureUnits object.
UnitsOfTemperature	String	The default units of temperature. Values are limited to the following strings: "DegreesCelcius" "DegreesFahrenheit"

Notes

1. Only one instance of the TemperatureUnits object is allowed in each file.

TimeUnits

Units Group
Product Category

Description Used to define the units of time.

Schema Entry

```
{TimeUnits
{Id TimeUnitsId}
{String UnitsOfTime}
}
```

Data Example

```
{TimeUnits 1 “Seconds”}
```

Attributes

Name	***Type***	***Description***
TimeUnitsId	Id	A unique number for each TimeUnits object.
UnitsOfTime	String	The default units of time. Values are limited to the following strings: “Centiseconds” “Hours” “Milliseconds” “Minutes” “Seconds”

Notes

1. Only one instance of the TimeUnits object is allowed in each file.

TorqueUnits

Units Group
Product Category

Description Used to define the units of torque.

Schema Entry

```
{TorqueUnits
{Id TorqueUnitsId}
{String UnitsOfTorque}
}
```

Data Example

{TorqueUnits 101 “NewtonMeters”}

Attributes

Name	***Type***	***Description***
TorqueUnitsId	Id	A unique number for each TorqueUnits object.
UnitsOfTorque	String	The default units of torque. Values are limited to the following strings: “FootPounds” “InchPounds” “NewtonMeters”

Notes

1. Only one instance of the TorqueUnits object is allowed in each file.

VelocityUnits

Units Group
Product Category

Description Used to define the units of velocity. Choices are limited to those indicated.

Schema Entry

```
{VelocityUnits
{Id VelocityUnitsId}
{String UnitsOfVelocity}
}
```

Data Example

```
{VelocityUnits 1 "Inches/Minute"}
```

Attributes

Name	*Type*	*Description*
VelocityUnitsId	Id	A unique number for each VelocityUnits object.
UnitsOfVelocity	String	The default units of velocity. Values are limited to the following strings: "Feet/Second" "Inches/Minute" "Inches/Second" "Meters/Second" "Millimeters/Minute" "Millimeters/Second"

Notes

1. Only one instance of the VelocityUnits object is allowed in each file.

VolumeUnits

Units Group
Product Category

Description Used to define the units of volume.

Schema Entry

```
{VolumeUnits
{Id VolumeUnitsId}
{String UnitsOfVolume}
}
```

Data Example

{VolumeUnits 1 “Microliters”}

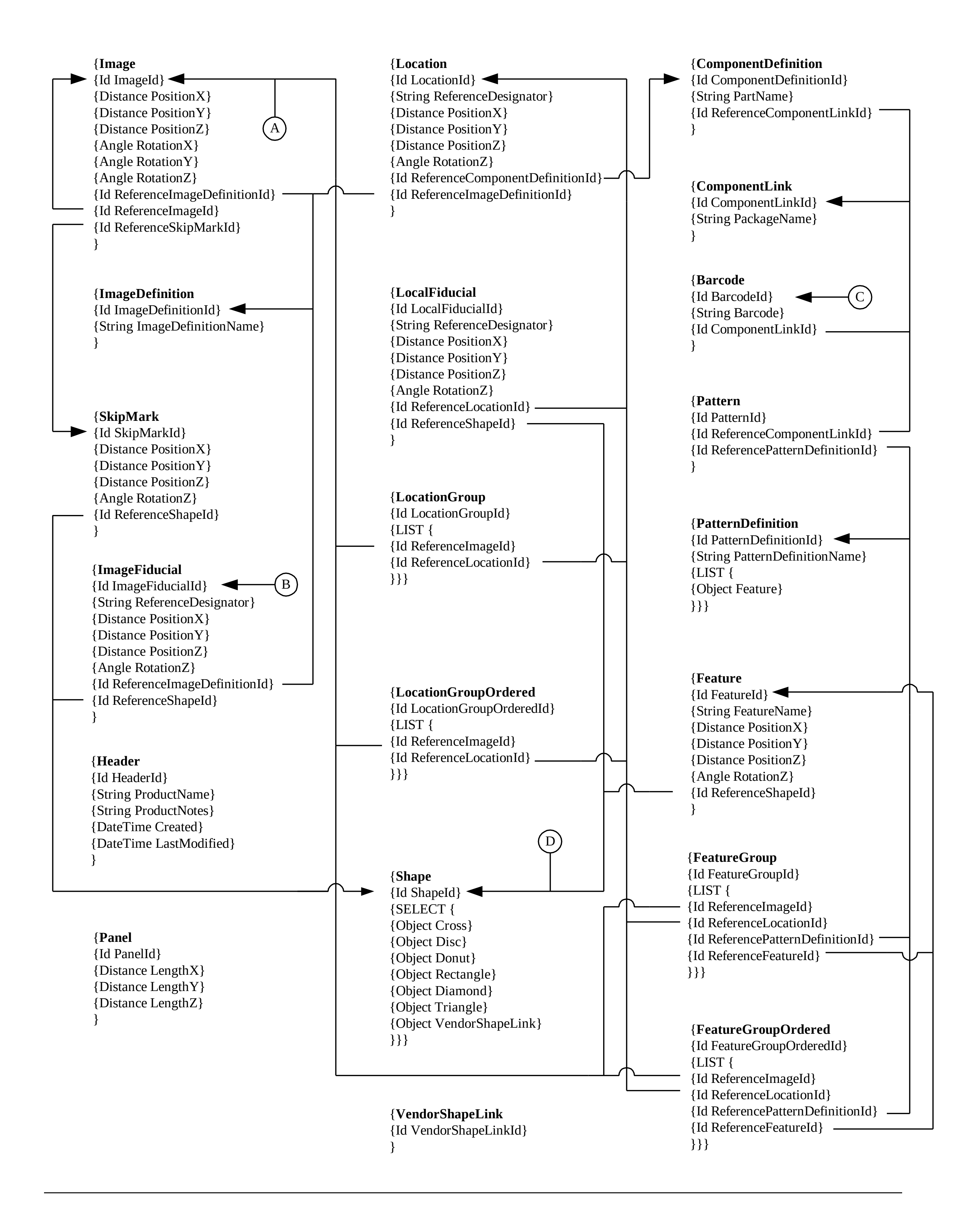
Attributes

Name	*Type*	*Description*
VolumeUnitsId	Id	A unique number for each VolumeUnits object.
UnitsOfVolume	String	The default units of volume. Values are limited to the following strings: “Feet^3” “FluidOunces” “Gallons” “Inches^3” “Liters” “Microliters” “Milliliters”

Notes

1. Only one instance of the VolumeUnits object is allowed in each file.

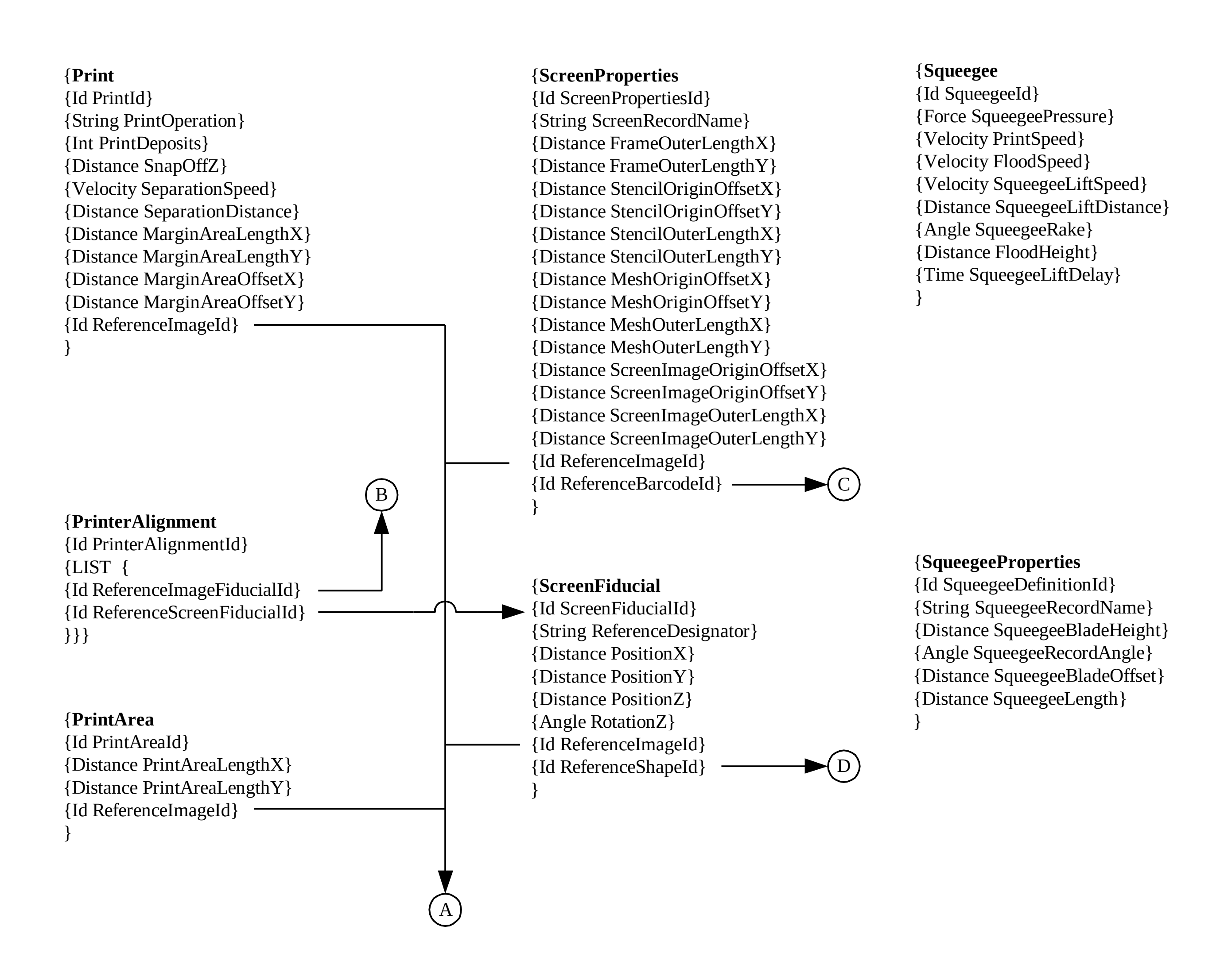
Appendix E
Entity Relationship Diagram


{Image
{Id ImageId}
{Distance PositionX}
{Distance PositionY}
{Distance PositionZ}
{Angle RotationX}
{Angle RotationY}
{Angle RotationZ}
{Id ReferenceImageDefinitionId}
{Id ReferenceImageId}
{Id ReferenceSkipMarkId}
}
A
{ImageDefinition
{Id ImageDefinitionId}
{String ImageDefinitionName}
}
{SkipMark
{Id SkipMarkId}
{Distance PositionX}
{Distance PositionY}
{Distance PositionZ}
{Angle RotationZ}
{Id ReferenceShapeId}
}
{ImageFiducial
{Id ImageFiducialId}
{String ReferenceDesignator}
{Distance PositionX}
{Distance PositionY}
{Distance PositionZ}
{Angle RotationZ}
{Id ReferenceImageDefinitionId}
{Id ReferenceShapeId}
}
B
{Header
{Id HeaderId}
{String ProductName}
{String ProductNotes}
{DateTime Created}
{DateTime LastModified}
}
{Panel
{Id PanelId}
{Distance LengthX}
{Distance LengthY}
{Distance LengthZ}
}
{Location
{Id LocationId}
{String ReferenceDesignator}
{Distance PositionX}
{Distance PositionY}
{Distance PositionZ}
{Angle RotationZ}
{Id ReferenceComponentDefinitionId}
{Id ReferenceImageDefinitionId}
}
{LocalFiducial
{Id LocalFiducialId}
{String ReferenceDesignator}
{Distance PositionX}
{Distance PositionY}
{Distance PositionZ}
{Angle RotationZ}
{Id ReferenceLocationId}
{Id ReferenceShapeId}
}
{LocationGroup
{Id LocationGroupId}
{LIST {
{Id ReferenceImageId}
{Id ReferenceLocationId}
}}}
{LocationGroupOrdered
{Id LocationGroupOrderedId}
{LIST {
{Id ReferenceImageId}
{Id ReferenceLocationId}
}}}
D
{Shape
{Id ShapeId}
{SELECT {
{Object Cross}
{Object Disc}
{Object Donut}
{Object Rectangle}
{Object Diamond}
{Object Triangle}
{Object VendorShapeLink}
}}}
{VendorShapeLink
{Id VendorShapeLinkId}
}
{ComponentDefinition
{Id ComponentDefinitionId}
{String PartName}
{Id ReferenceComponentLinkId}
}
{ComponentLink
{Id ComponentLinkId}
{String PackageName}
}
{Barcode
{Id BarcodeId}
{String Barcode}
{Id ComponentLinkId}
}
C
{Pattern
{Id PatternId}
{Id ReferenceComponentLinkId}
{Id ReferencePatternDefinitionId}
}
{PatternDefinition
{Id PatternDefinitionId}
{String PatternDefinitionName}
{LIST {
{Object Feature}
}}}
{Feature
{Id FeatureId}
{String FeatureName}
{Distance PositionX}
{Distance PositionY}
{Distance PositionZ}
{Angle RotationZ}
{Id ReferenceShapeId}
}
{FeatureGroup
{Id FeatureGroupId}
{LIST {
{Id ReferenceImageId}
{Id ReferenceLocationId}
{Id ReferencePatternDefinitionId}
{Id ReferenceFeatureId}
}}}
{FeatureGroupOrdered
{Id FeatureGroupOrderedId}
{LIST {
{Id ReferenceImageId}
{Id ReferenceLocationId}
{Id ReferencePatternDefinitionId}
{Id ReferenceFeatureId}
}}}

{**DispenseOrder**
{Id DispenseOrderId}
{LIST {
{SELECT {
{Object FeatureGroup}
{Object FeatureGroupOrdered}
}}}}}

{**InspectOrder**
{Id InspectOrderId}
{LIST {
{SELECT {
{Object FeatureGroup}
{Object FeatureGroupOrdered}
}}}}}

{**PlacementOrder**
{Id PlacementOrderId}
{LIST {
{SELECT {
{Object LocationGroup}
{Object LocationGroupOrdered}
}}}}}



{**Header**
{Id HeaderId}
{String ProductName}
{String ProductNotes}
}

{**SRFFVersion**
{Id SRFFVersionId}
{String VersionName}
}

{**AccelerationUnits**
{Id AccelerationUnitsId}
{String UnitsOfAcceleration}
}

{**AngleUnits**
{Id AngleUnitsId}
{String UnitsOfAngle}
}

{**AngularVelocityUnits**
{Id AngularVelocityId}
{String
UnitsOfAngularVelocity}
}

{**AngularAccelerationUnits**
{Id AngularAccelerationId}
{String UnitsOfAngularAcceleration}
}

{**DistanceUnits**
{Id DistanceUnitsId}
{String UnitsOfDistance}
}

{**FlowUnits**
{Id FlowUnitsId}
{String UnitsOfFlow}
}

{**ForceUnits**
{Id ForceUnitsId}
{String UnitsOfForce}
}

{**HumidityUnits**
{Id HumidityUnitsId}
{String UnitsOfHumidity}
}

{**MassUnits**
{Id MassUnitsId}
{String UnitsOfMass}
}

{**PowerUnits**
{Id PowerUnitsId}
{String UnitsOfPower}
}

{**PressureUnits**
{Id PressureUnitsId}
{String UnitsOfPressure}
}

{**TemperatureUnits**
{Id TemperatureUnitsId}
{String UnitsOfTemperature}
}

{**TimeUnits**
{Id TimeUnitsId}
{String UnitsOfTime}
}

{**TorqueUnits**
{Id TorqueUnitsId}
{String UnitsOfTorque}
}

{**VelocityUnits**
{Id VelocityUnitsId}
{String UnitsOfVelocity}
}

{**VolumeUnits**
{Id VolumeUnitsId}
{String UnitsOfVolume}
}

{**Cross**
{Id CrossId}
{Distance Base}
{Distance Height}
{Distance BaseLegWidth}
{Distance HeightLegWidth}
}

{**Diamond**
{Id DiamondId}
{Distance Base}
{Distance Height}
}

{**Disc**
{Id DiscId}
{Distance Diameter}
}

{**Donut**
{Id DonutId}
{Distance InnerDiameter}
{Distance OuterDiameter}
}

{**Rectangle**
{Id RectangleId}
{Distance Base}
{Distance Height}
}

{**Triangle**
{Id TriangleId}
{Distance Base}
{Distance Height}
{Distance Offset}
}

Appendix F
Coordinate Systems Graphics
and Definitions

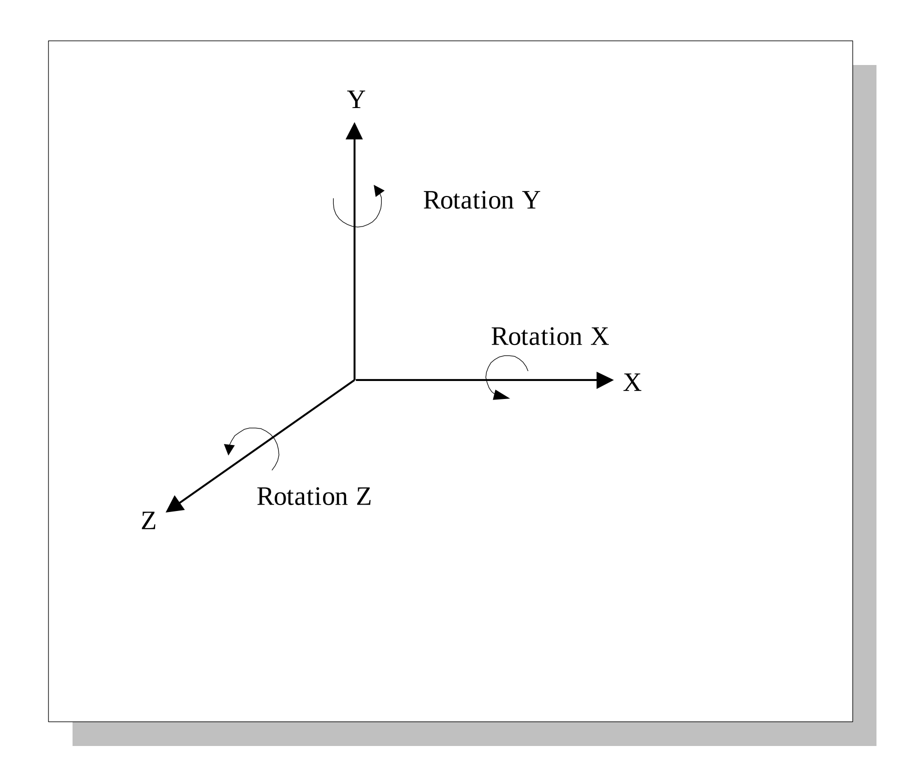


Figure F.1 Coordinate system conventions to be used in SRFF files.

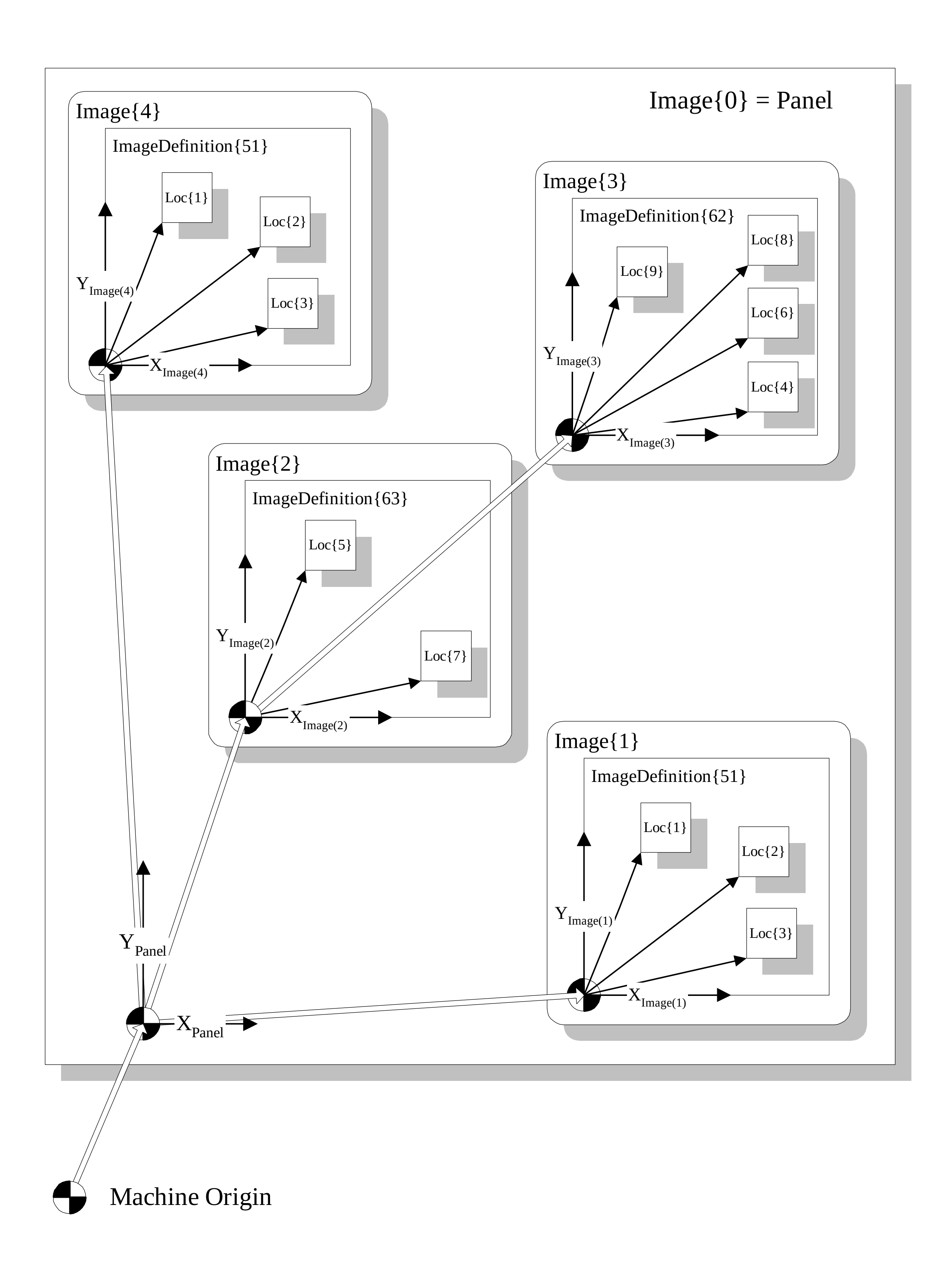


Figure F.2 An example indicating the spatial relationships among a machine origin, a Panel, four Images (five counting the panel), three ImageDefinitions, and various Locations.

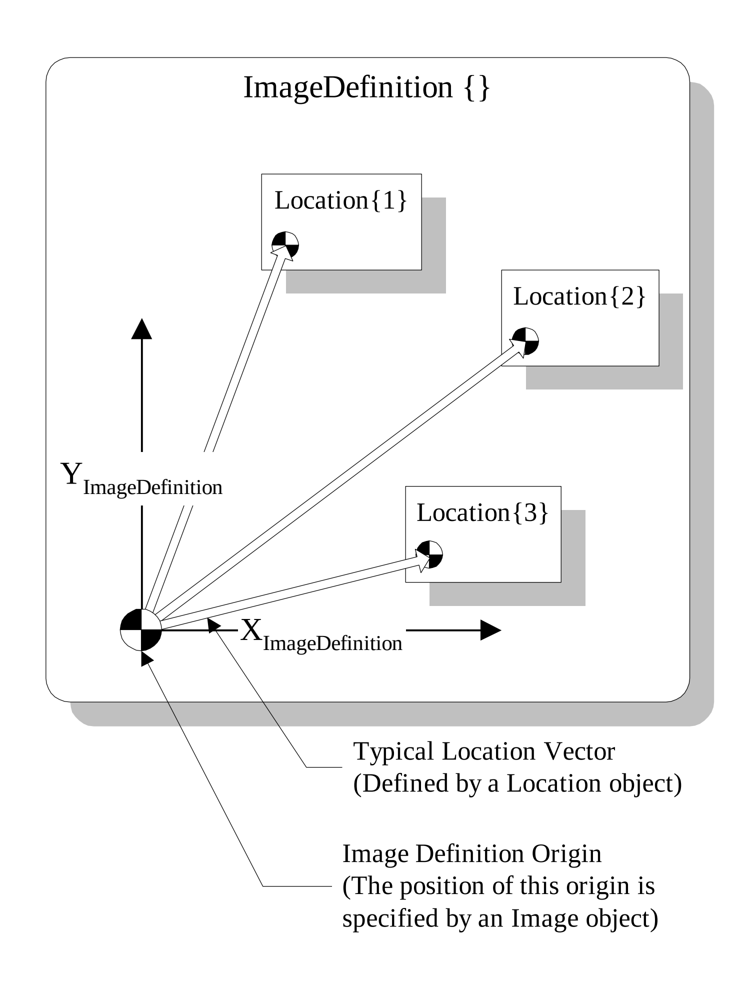


Figure F.3 An ImageDefinition and associated Locations

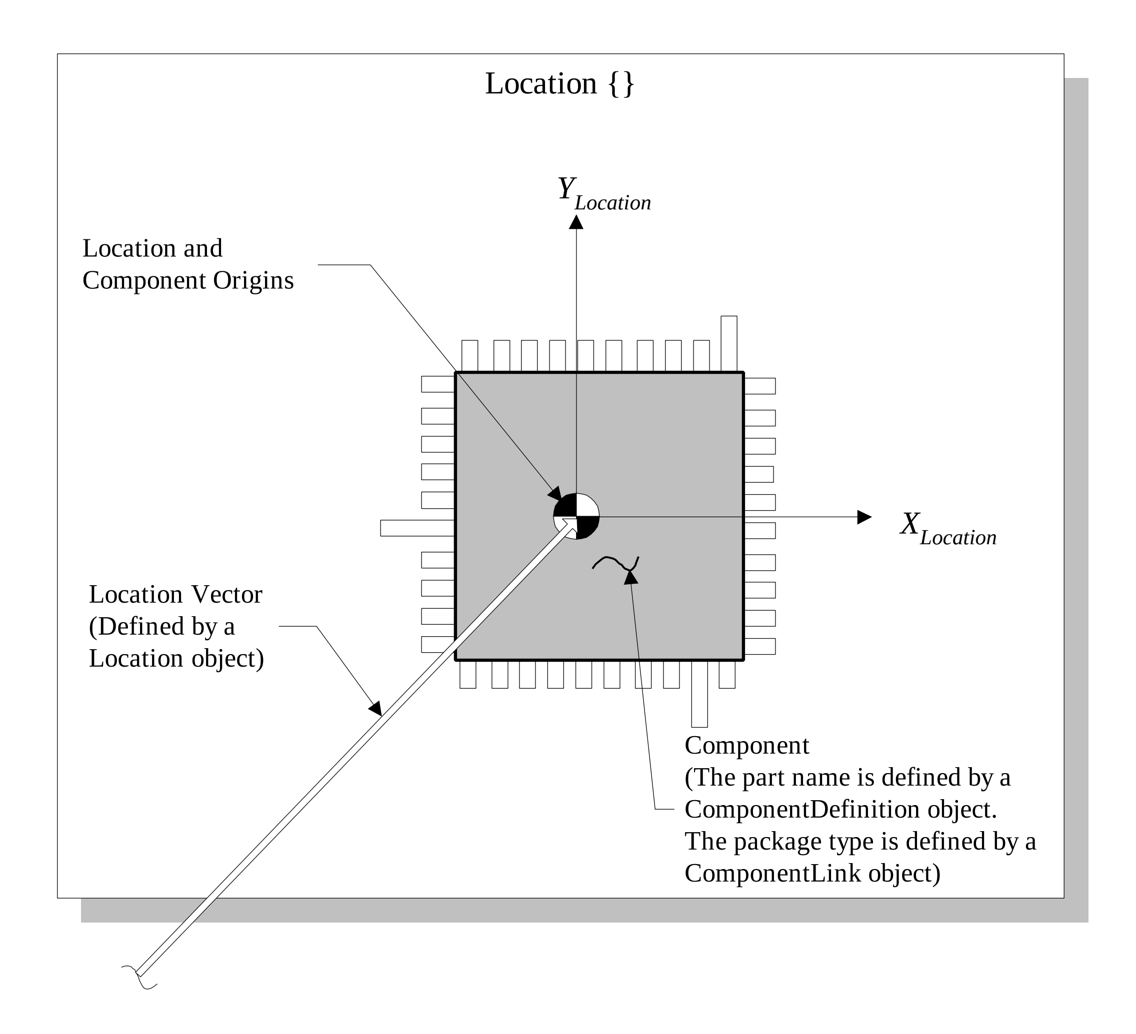


Figure F.4 A Location with corresponding component. The Location coordinate system and the component coordinate system are defined to be synonymous.

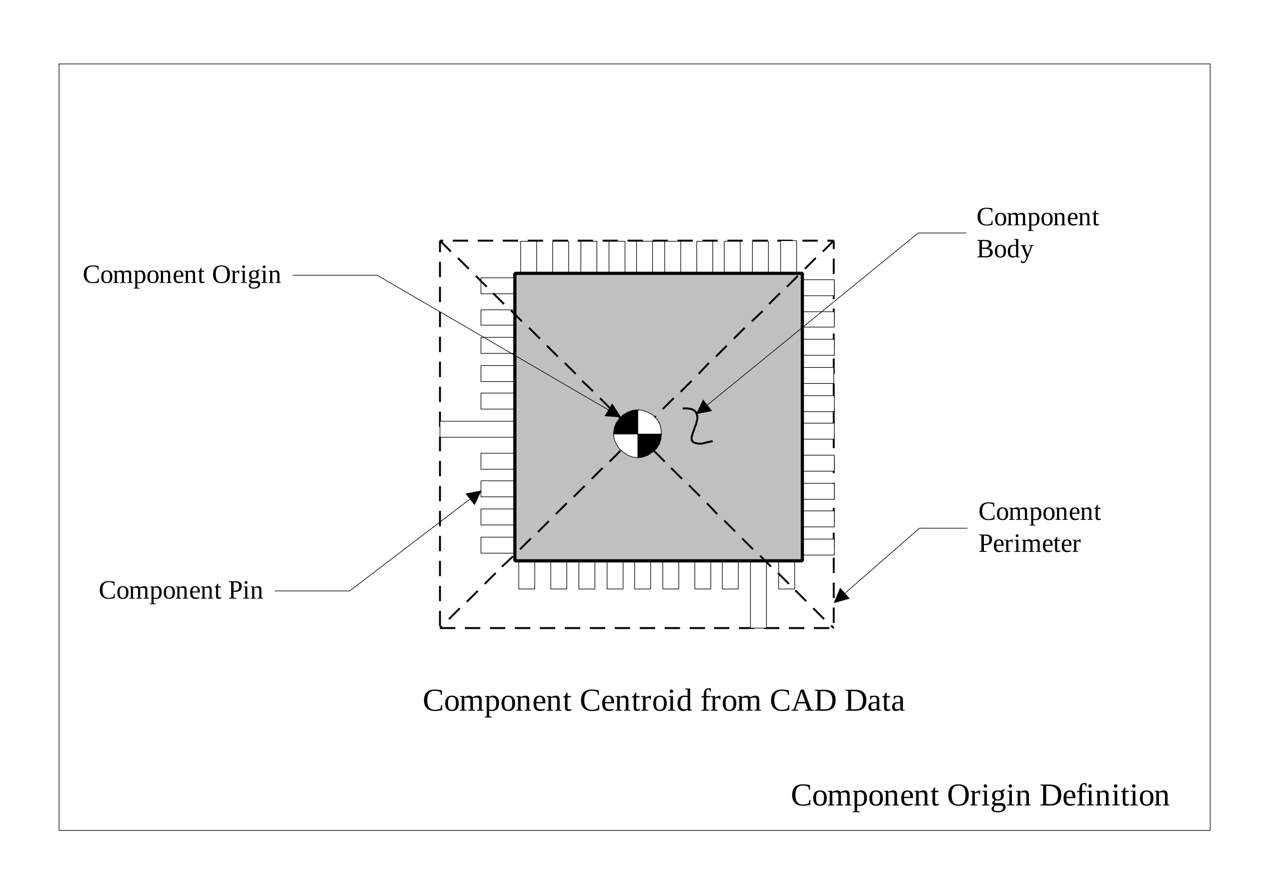


Figure F.5 Definition of a component origin.

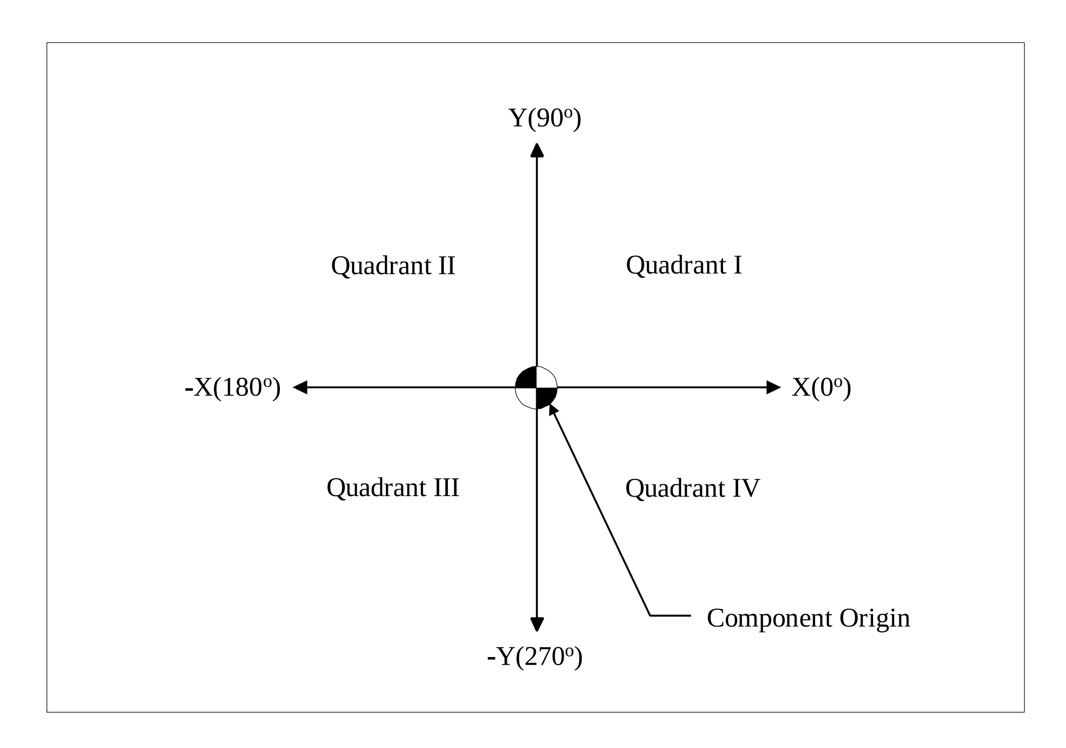


Figure F.6 Quadrants used for determining the orientation of a component.

When viewed from the top of a component, the component is oriented at:

0º	if Pin 1 of the component lies in Quadrant II
90º	if Pin 1 of the component lies in Quadrant III
180º	if Pin 1 of the component lies in Quadrant IV
270º	if Pin 1 of the component lies in Quadrant I

where,

0º	<	Quadrant I	<=	90º
90º	<	Quadrant II	<=	180º
180º	<	Quadrant III	<=	270º
270º	<	Quadrant IV	<=	360º

Note: If Pin 1 of the component lies on the location origin, then Pin 2 is used to determine orientation.

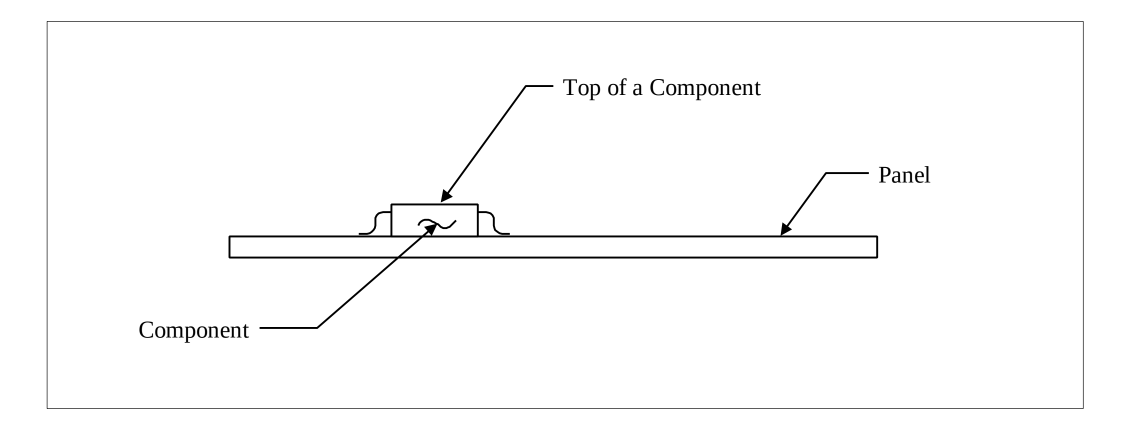


Figure F.7 Convention used to determine the top of a component.

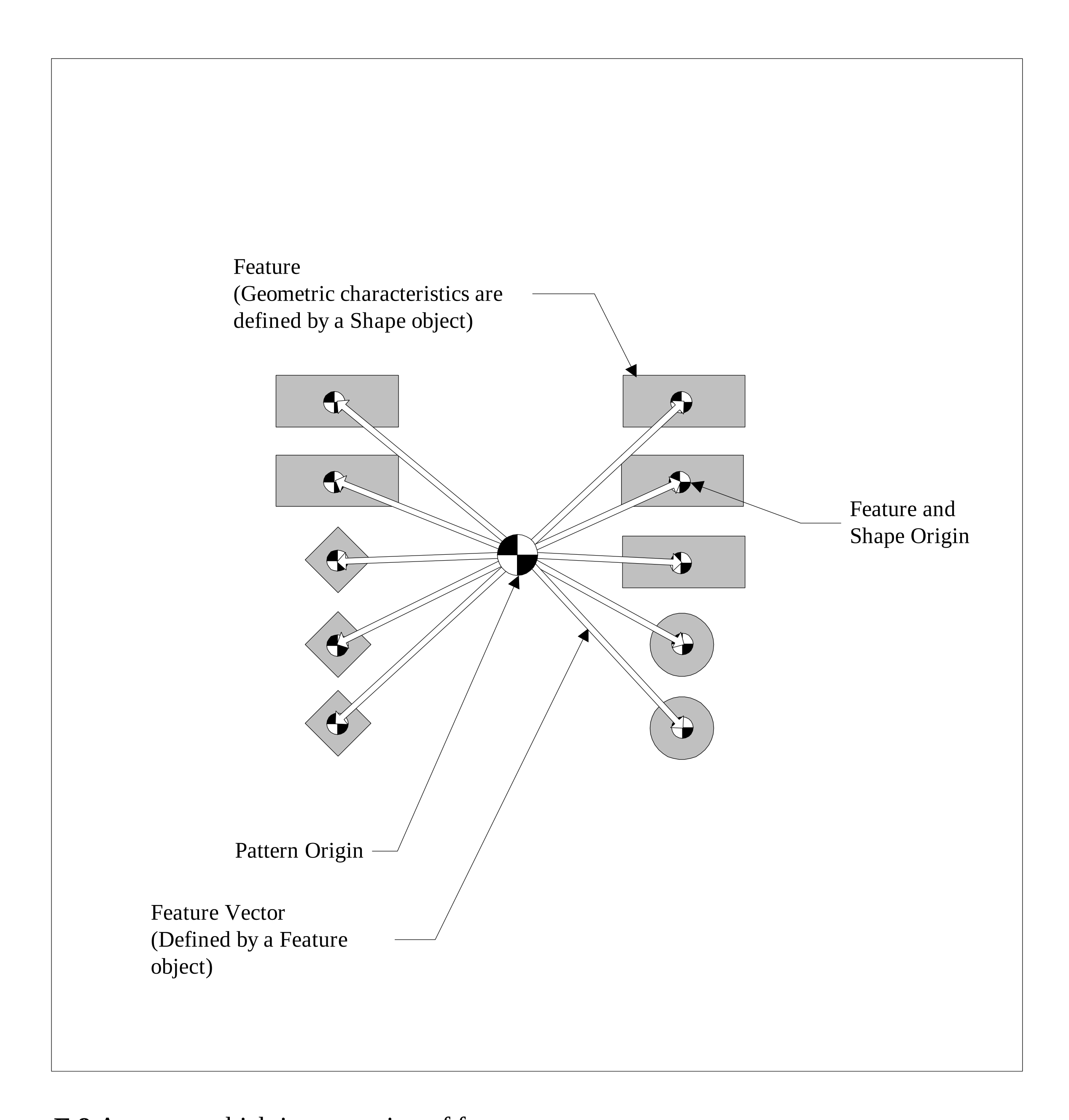


Figure F.8 A pattern which is a grouping of features.

Appendix G
File Example

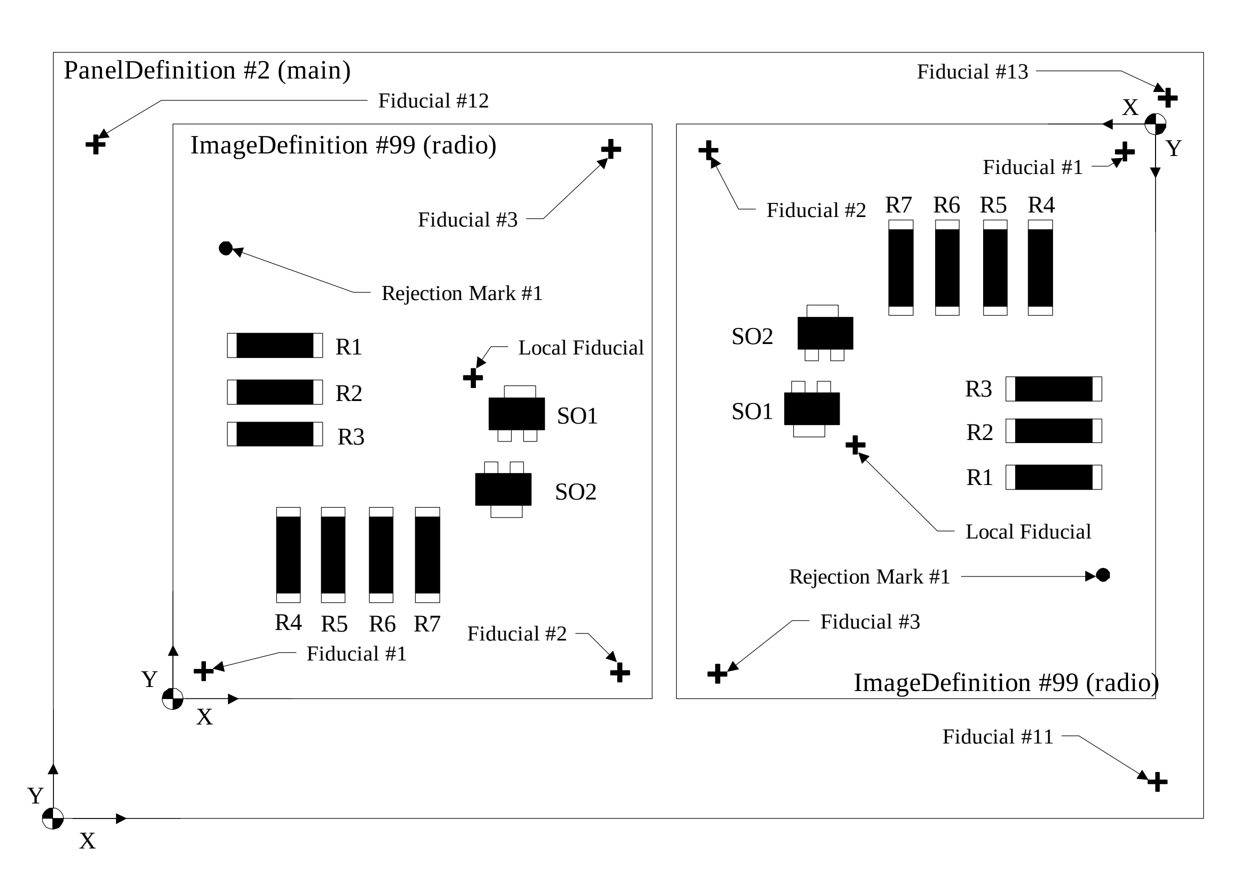


Figure G.1, Example used in this Appendix.

```
#---------------------------------------------
#     Begin File
#---------------------------------------------

#
#     Please note:
#     This file may have errors.  It is supplied for reference only.
#

      #---------------------------------------------
      #     Begin Schema
      #---------------------------------------------
      {Schema

            #---------------------------------------------
            #     Begin Product Schema Section
            #---------------------------------------------
            {Product

                  #---------------------------------------------
                  #     Begin SMEMA Product Schema Section
                  #---------------------------------------------
                  {Organization SMEMA

{ComponentDefinition
{Id         ComponentDefinitionId}
{String     PartName}
{Id         ReferenceComponentLinkId}
}

{ComponentLink
{Id         ComponentLinkId}
{String     PackageName}
}

{Feature
{Id         FeatureId}
{String     FeatureName}
{Distance   PositionX}
{Distance   PositionY}
{Distance   PositionZ}
{Angle      RotationZ}
{Id         ReferenceShapeId}
}

{Header
{Id         HeaderId}
{String     ProductName}
{String     ProductNotes}
{DateTime   Created}
{DateTime   LastModified}
}



{Image
{Id         ImageId}
```

```
{Distance   PositionX}
{Distance   PositionY}
{Distance   PositionZ}
{Angle      RotationX}
{Angle      RotationY}
{Angle      RotationZ}
{Id         ReferenceImageDefinitionId}
{Id         ReferenceImageId}
{Id         ReferenceSkipMarkId}
}

{ImageDefinition
{Id         ImageDefinitionId}
{String     ImageDefinitionName}
}

{ImageFiducial
{Id         ImageFiducialId}
{String     ReferenceDesignator}
{Distance   PositionX}
{Distance   PositionY}
{Distance   PositionZ}
{Angle      RotationZ}
{Id         ReferenceImageDefinitionId}
{Id         ReferenceShapeId}
}

{LocalFiducial
{Id         LocalFiducialId}
{String     ReferenceDesignator}
{Distance   PositionX}
{Distance   PositionY}
{Distance   PositionZ}
{Angle      RotationZ}
{ID         ReferenceLocationId}
{ID         ReferenceShapeId}
}

{Location
{Id         LocationId}
{String     ReferenceDesignator}
{Distance   PositionX}
{Distance   PositionY}
{Distance   PositionZ}
{Angle      RotationZ}
{Id         ReferenceComponentDefinitionId}
{Id         ReferenceImageDefinitionId}
}



{Panel
{Id         PanelId}
{Distance   LengthX}
{Distance   LengthY}
{Distance   LengthZ}
}
```

```
{Pattern
{Id         PatternId}
{String     PatternName}
{Id         ReferenceComponentLinkId}
{Id         ReferencePatternDefinitionId}
}

{PatternDefinition
{Id         PatternDefinitionId}
{String     PatternDefinitionName}
{LIST {
{Object     Feature}
}}}

{Shape
{Id         ShapeId}
{SELECT     {
{Object     Cross}
{Object     Disc}
{Object     Diamond}
{Object     Donut}
{Object     Rectangle}
{Object     Diamond}
{Object     Triangle}
{Object     VendorShapeLink}
}}}

{SkipMark
{Id         SkipMarkId}
{Distance   PositionX}
{Distance   PositionY}
{Distance   PositionZ}
{Angle      RotationZ}
{Id         ReferenceShapeId}
}

{SRFFVersion
{Id         SRFFVersionId}
{String     VersionName}
}

{VendorShapeLink
{Id         VendorShapeLinkId}
}



{Cross
{Id         CrossId}
{Distance   Base}
{Distance   Height}
{Distance   BaseLegWidth}
{Distance   HeightLegWidth}
}

{Diamond
{Id         DiamondId}
```

```
{Distance   Base}
{Distance   Height}
}

{Disc
{Id         DiscId}
{Distance   Diameter}
}

{Donut
{Id         DonutId}
{Distance   InnerDiameter}
{Distance   OuterDiameter}
}

{Rectangle
{Id         RectangleId}
{Distance   Base}
{Distance   Height}
}

{Triangle
{Id         TriangleId}
{Distance   Base}
{Distance   Height}
{Distance   Offset}
}

{AngleUnits
{Id         AngleUnitsId}
{String     UnitsOfAngle}
}

{DistanceUnits
{Id         DistanceUnitsId}
{String     UnitsOfDistance}
}

{TimeUnits
{Id         TimeUnitsId}
{String     UnitsOfTime}
}

                  #---------------------------------------------
                  #     End SMEMA Product Schema Section
                  #---------------------------------------------
                  }

                  #---------------------------------------------
                  #     Begin Vendor Product Schema Section
                  #---------------------------------------------
                  {Organization Vendor

{Fiducial
{Id         FiducialId}
{Id         ShapeId}
{String     BinData}
```

```
}

                  #---------------------------------------------
                  #     End Vendor Product Section
                  #---------------------------------------------
                  }

            #---------------------------------------------
            #     End Product Schema Section
            #---------------------------------------------
            }

            #---------------------------------------------
            #     Begin Process Schema Section
            #---------------------------------------------
            {Process

                  #---------------------------------------------
                  #     Begin SMEMA Process Schema Section
                  #---------------------------------------------
                  {Organization SMEMA

{FeatureGroup
{Id         FeatureGroupId}
{LIST       {
{Id         ReferenceImageId}
{Id         ReferenceLocationId}
{Id         ReferrencePatternDefinitonId}
{Id         ReferenceFeatureId}
}}}

{FeatureGroupOrdered
{Id         FeatureGroupOrderedId}
{LIST       {
{Id         ReferenceImageId}
{Id         ReferenceLocationId}
{Id         ReferrencePatternDefinitionId}
{Id         ReferenceFeatureId}
}}}

{LocationGroup
{Id         LocationGroupId}
{LIST       {
{Id         ReferenceImageId}
{Id         ReferenceLocationId}
}}}

{LocationGroupOrdered
{Id         LocationGroupOrderedId}
{LIST       {
{Id         ReferenceImageId}
{Id         ReferenceLocationId}
}}}

{PlacementOrder
{Id         PlacementOrderId}
{LIST       {
{SELECT     {
```

```
{Object             LocationGroup}
{Object             LocationGroupOrdered}
}}}}}

                        #---------------------------------------------
                        #     End SMEMA Process Schema Section
                        #---------------------------------------------
                        }

                        #---------------------------------------------
                        #     Begin Vendor Process Schema Section
                        #---------------------------------------------
                        {Organization Vendor

{Dot
{Id          DotId}
{Id          Valve}
{Time        ShotSize}
{Distance    UpHeight}
{Distance    DownHeight}
{Time        Dwell}
{Distance    ShiftUpHeight}
{Distance    ShiftDownHeight}
{Distance    ToggleHeight}
{Bool        DispenseOnDown}
{Bool        DispenseOnUp}
}

                        #---------------------------------------------
                        #     End Vendor Process Schema Section
                        #---------------------------------------------
                        }



            #---------------------------------------------
            #     End Process Schema Section
            #---------------------------------------------
            }

      #---------------------------------------------
      #     End Schema Section
      #---------------------------------------------
      }

      #---------------------------------------------
      #     Begin Data Section
      #---------------------------------------------
      {Data

            #---------------------------------------------
            #     Begin Product Section
            #---------------------------------------------
            {Product

                  #---------------------------------------------
                  #     Begin SMEMA Product Data Section
```

```
                  #---------------------------------------------
                  {Organization SMEMA

{Header 1
            "SRFF Example 001"
            "SMEMA SRFF Specification Appendix G"
            "1996-07-12T02:32.45.00+01:00"
            "1998-02-23T16:13:08.00-05:00"
}

{SRFFVersion 1 "1.0"}
{DistanceUnits 1 "Microns"}
{AngleUnits 1 "Degrees/1000"}
{Panel 2 119000 149000 *}
{Image 0 0 0 0 0 0 0 * * 1}
{Image 1 13000 19000 0 0 0 0 98 0 1}
{Image 2 108000 142000 0 180 0 0 99 0 1}
{ImageDefinition 98 "panel"}
{Location 1 "R1" 15000 54000 0 0 98 1}
{Location 2 "R2" 15000 48000 0 0 98 1}
{Location 3 "R3" 15000 42000 0 0 98 2}
{Location 4 "R4" 16500 23000 0 90000 98 3}
{Location 5 "R5" 22500 23000 0 90000 98 3}
{Location 6 "R6" 28500 23000 0 90000 98 4}
{Location 7 "R7" 34500 23000 0 90000 98 4}
{Location 8 "SO1" 45000 46000 0 90000 98 5}
{Location 9 "SO2" 45000 33000 0 27000 98 5}
{ImageDefinition 99 "radio"}
{Location 1 "R1" 15000 54000 0 0 99 1}
{Location 2 "R2" 15000 48000 0 0 99 1}
{Location 3 "R3" 15000 42000 0 0 99 2}
{Location 4 "R4" 16500 23000 0 90000 99 3}
{Location 5 "R5" 22500 23000 0 90000 99 3}
{Location 6 "R6" 28500 23000 0 90000 99 4}
{Location 7 "R7" 34500 23000 0 90000 99 4}
{Location 8 "SO1" 45000 46000 0 90000 99 5}
{Location 9 "SO2" 45000 33000 0 27000 99 5}
{ComponentDefinition 1 "412-13763" 1}
{ComponentDefinition 2 "412-13812" 1}
{ComponentDefinition 3 "412-76421" 1}
{ComponentDefinition 4 "412-76423" 1}
{ComponentDefinition 5 "549-32499" 2}
{ComponentLink 1 "1206"}
{ComponentLink 2 "SOT"}
{SkipMark 1 7000 70500 0 0 254}
{Shape 254 {Disc 1 3000}}
{Shape 255 {Cross 1 3000 3000 1000 1000}}
{ImageFiducial 11 "IF1" 142000 4000 0 0 98 255}
{ImageFiducial 12 "IF2" 5000 105000 0 0 98 255}
{ImageFiducial 13 "IF3" 144000 112000 0 0 98 255}
{ImageFiducial 1 "IF4" 3000 3000 0 0 99 255}
{ImageFiducial 2 "IF5" 58500 4000 0 0 99 255}
{ImageFiducial 3 "IF6" 57000 84500 0 0 99 255}
{Pattern 1 "Pad Pattern for 256 QFP" 1 2}
{Pattern 2 "Pad Pattern" 2 1}
{PatternDefinition 1 "Glue" {
{Feature 1 "Dot1" 1000 1000 0 0 254}
{Feature 2 "Dot2" 1000 1000 0 0 254}
```

```
{Feature 3 "Dot3" 1000 1000 0 0 254}
{Feature 4 "Dot4" 1000 1000 0 0 254}}
{PatternDefinition 2  "Solder" {
      {Feature 5 "Solder1" 100 0 0 0 254}
      {Feature 6 "Solder2" 1400 0 0 0 254}
}

                  #--------------------------------------------
                  #     End SMEMA Product Data Section
                  #--------------------------------------------
                  }

                  #--------------------------------------------
                  #     Begin Vendor Product Data Section
                  #--------------------------------------------
                  {Organization Vendor

{Fiducial 1 255
“
begin 600 UUCODE.TXT
M06YD<F5W($1U9V5N<VME#0I-86YU9F%C='5R:6YG(%)E<V5A<F-H($-E;G1E
M<@T*1V5O<F=I82!);G-T:71U=&4@;V8@5&5C:&YO;&]G>0T*.#$S($9E<G-T
M($1R:79E#0I!=&QA;G1A+"!'02`S,#,S,BTP-3`V,`T**#0P-"D@.#DT+3DQ
M-C$-"B@T,#0I(#@Y-"TP.34W($9A>`T*86YD<F5W+F1U9V5N<VME0&UA<F,N
M9V%T96-H+F5D=0T*:'1T<#HO+W=W=RYD=6=E;G-K92YM87)C+F=A=&5C:"YE
"9'5T
`
end
``
                  #---------------------------------------------
                  #     End Vendor Product Data Section
                  #--------------------------------------------
                  }

            #--------------------------------------------
            #     End Product Data Section
            #--------------------------------------------
            }

            #--------------------------------------------
            #     Begin Process Data Section
            #--------------------------------------------
            {

                  #---------------------------------------------
                  #     Begin SMEMA Process Data Section
                  #---------------------------------------------
                  {Organization SMEMA

{PlacementOrder 1 {LocationGroupOrdered 1 {1 8 2 8 1 9 2 9}}
{LocationGroup 2 {1 3 1 4 1 5 1 6 1 7 2 3 2 4 2 5 2 6 2 7}}

                  #--------------------------------------------
                  #     End SMEMA Process Data Section
                  #---------------------------------------------
                  }

                  #--------------------------------------------
```

```
                  #     Begin Vendor Process Data Section
                  #------------------------------------------
                  {Organization Vendor

{Dot 1 1 30 2032 0 10 0 0 0 True False}
{Dot 2 1 25 1900 0 10 0 0 0 True False)

                  #------------------------------------------
                  #     End Vendor Process Data Section
                  #------------------------------------------
                  }

            #------------------------------------------
            #     End Process Data Section
            #------------------------------------------
            }

      #------------------------------------------
      #     End Data Section
      #------------------------------------------
      }

#------------------------------------------
#     End File
#------------------------------------------
```

Appendix H
Error Codes

Table H.1

Code	***Error***	***Description***
SRFF 1001	Missing Schema	The schema portion of the file is missing or was not detected.
SRFF 1002	Invalid Schema	The schema does not match the BNF grammar.
SRFF 1003	Duplicated Schema	Two or more schema sections are present.
SRFF 1011	Invalid Product Schema	Structure does not match BNF grammar.
SRFF 1012	Duplicate Product Schema	Two or more product schema sections are present.
SRFF 1013	Invalid SMEMA Product Schema	The SMEMA product schema does not match the BNF grammar.
SRFF 1014	Duplicate SMEMA Product Schema	Two or more SMEMA product schema sections are present.
SRFF 1015	Invalid Vendor Product Schema	A vendor product schema does not match the BNF grammar.
SRFF 1021	Invalid Process Schema	Structure does not match the BNF grammar.
SRFF 1022	Duplicate Process Schema Section	Two or more process schema sections are present.
SRFF 1023	Invalid SMEMA Process Schema	The SMEMA process schema does not match the BNF grammar.
SRFF 1024	Duplicate SMEMA Process Schema	Two or more SMEMA process schema sections are present.
SRFF 1025	Invalid Vendor Process Section	A vendor process schema does not match the BNF grammar.
SRFF 2001	Missing Data Section	The data section is missing.
SRFF 2002	Invalid Data Section	The data section does not match the BNF grammar.
SRFF 2003	Duplicate Data Section	Two or more data sections are present.
SRFF 2011	Invalid Product Data Section	Structure does not match the BNF grammar.
SRFF 2012	Duplicate Product Data Section	Two or more product data sections are present.
SRFF 2013	Invalid SMEMA Product Data Section	The SMEMA product data section does not match the BNF grammar.
SRFF 2014	Duplicate SMEMA Product Data Section	Two or more SMEMA product data sections are present.
SRFF 2015	Invalid Vendor Product Data Section	A vendor product data section does not match the BNF grammar.
SRFF 2021	Invalid Process Data Section	Structure does not match the BNF grammar.
SRFF 2022	Duplicate Process Data Section	Two or more process data sections are present.
SRFF 2023	Invalid SMEMA Process Data Section	The SMEMA process data section does not match the BNF grammar.
SRFF 2024	Duplicate SMEMA Process Data Section	Two or more SMEMA process data sections are present.
SRFF 2025	Invalid Vendor Process Data Section	A vendor process data section does not match the BNF grammar.

Table H.1 Continued

SRFF 3001	Invalid Object Name	The object name contains illegal characters, or starts with an illegal character.
SRFF 3002	Object Name Too Long	The object name is greater than 64 characters.
SRFF 3003	Invalid Attribute Name	The attribute name contains illegal characters, or starts with an illegal character.
SRFF 3004	Attribute Name Too Long	The attribute name is greater than 64 characters.
SRFF 3005	Invalid Attribute	An Id is missing from the object definition.
SRFF 3005	Illegal Use Of Reserved Word	
SRFF 3006	Unknown Data Type	Data type indicated is not supported.
SRFF 3007	Left Bracket Missing	
SRFF 3008	Right Bracket Missing	
SRFF 3009	Invalid Character Detected	A nonASCII character is included.
SRFF 3010	Extraneous Character	An extra character (an extra quote, an extra letter) is present.
SRFF 4001	Object Multiply Defined In Schema	Two or more definitions of the same object are present.
SRFF 4002	No Schema For This Vendor Data Section	There is not a corresponding schema for a detected vendor data section.
SRFF 4003	Unrecognized Object In Data Section	Object in data was not defined in schema
SRFF 4004	Too Many Attributes	
SRFF 4005	Too Few Attributes	
SRFF 5001	Invalid Data Type	Attempted to assign the wrong data type to an attribute.
SRFF 5002	Invalid Value For Unit Object Attribute	Only certain strings can be used to define the units to be used in a Unit Object.
SRFF 5003	Multiple Instance Of Same Unit Object	Only one instance of each unit object can be contained in a file.
SRFF 5004	ID Replicated For The Same Object	
SRFF 5005	Invalid Selection From Select Structure	Something was assigned as an attribute value that was not listed as a potential selection candidate in the schema.
SRFF 5006	Feature Not Found In Pattern	
SRFF 5007	Reference ID Unknown	An object is referencing an object that doesn't appear to exist.
SRFF 5008	Image 0 Missing	Image 0 is not present when other images are present.

Appendix I
Vendor Specific Object Naming Form

Vendor Specific Object Naming Form

Name		***Vendor Tag***	
Release		***Product or Process***	

Description

Schema Entry

Data Example

Attributes

Name	***Type***	***Description***

Notes

Appendix J
Compliance Forms

SRFF Equipment Compliance Form

SMEMA Vendor Name [1]	
Equipment Model	

Performance Level

Level	***Requirement***	***SRFF Version***	***Check Box v = yes***
1	The compliance statement is included in documentation supplied with the equipment. The documentation contains a list of standard error codes supported by the equipment.		
1	The equipment can accept and understand the contents of an SRF file. This includes accepting all common objects and equipment specific objects for the category of equipment. The equipment can perform process actions based on the information in the SRF file.		
2	The equipment has achieved compliance to Level 1 and: Can edit or create a file at the equipment user interface and save the file in SRF format retaining the information required for the specific equipment.		

Equipment Category

Category	***Check Box v = yes***
Dispense	
Inspect	
Material Movement	
Placement	
Print	
Reflow	
Test	
Wave Solder	
Other (Specify)	

Equipment Usage of Data Objects

Category	***Check Box v = yes***
Common Product Objects	
Common Process Objects	
Vendor Specific Product Objects	
Vendor Specific Process Objects	

[1] Registration of the Vendor name with SMEMA is mandatory to achieve compliance.

SRFF Software Compliance Form

SMEMA Vendor Name[1]	
Software Tool Name	

Capabilities

Vendor	***Equipment Model***	***Creation v = yes***	***Translation v = yes***	***Editing v = yes***	***Syntax v = yes***

Definitions

Creation
The software tool can create an SRFF file for at least 1 category of equipment. The documentation for the tool should identify manufacturers by name and identify specific equipment models.

Translation
The software tool can translate product information from a non-SRFF format to SRFF format. The documentation should include information about the formats supported by the tool.

Editing
The software tool allows a user to view, amend, and save information to SRFF file.

Syntax Check
The software tool can perform a syntax check of an SRFF format file. This includes checking the contents of the file against the SRFF grammar and checking that the Data section of the file complies with the Schema section. Errors are reported using the standard error codes. The documentation should include a list of the standard error codes supported by the tool.

Appendix K
Method to Register Company Name with SMEMA

To be determined by SMEMA